RasPi

DESIGN
BUILD
CODE

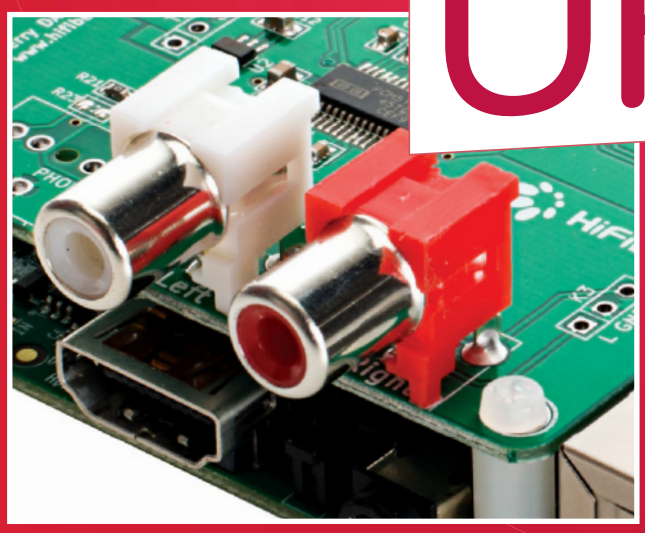
22

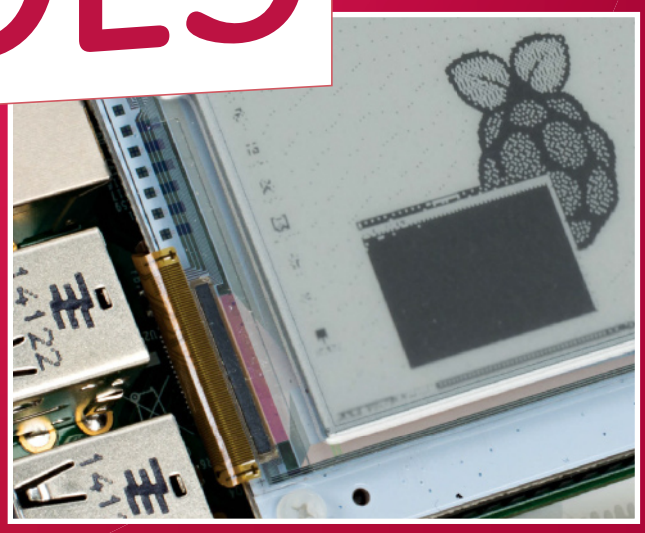
Get hands-on with your Raspberry Pi

HOW TO
POWER
UP YOUR PI

10

AWESOME RASPBERRY PI UPGRADES





Plus Use an Android device as a screen

Welcome

Your Raspberry Pi is a wonderful device, but you can make it even better. From infrared vision to pan-and-tilt camera control, there are a host of add-ons for the Pi that enable you to do almost anything you want. Our roundup of ten awesome upgrades demonstrates just some of what's possible.

Amiibos have blurred the crossover between the physical world and some of our favourite videogames, and in this issue you'll discover how one enthusiast made his own NFC-enabled papercraft cubes to bring the same experience to Minecraft. Make your own and marvel at how you can use them to build in-game. Plus we'll teach you how to power up your Pi, make a visual novel and more.

April

Editor

Get inspired

Discover the RasPi community's best projects

Expert advice

Got a question? Get in touch and we'll give you a hand

Easy-to-follow guides

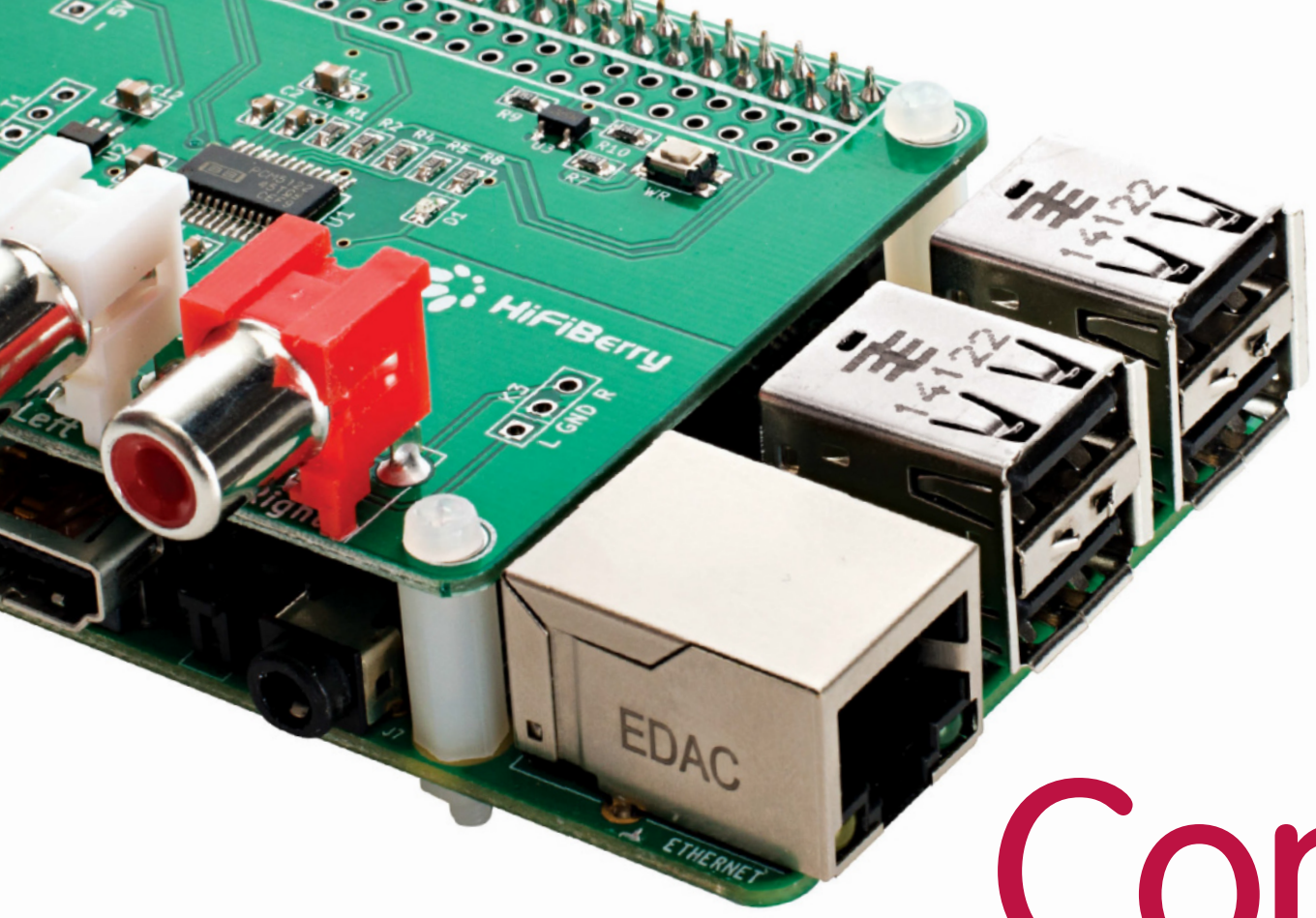
Learn to make and code gadgets with Raspberry Pi

From the makers of

LinuxUser & Developer

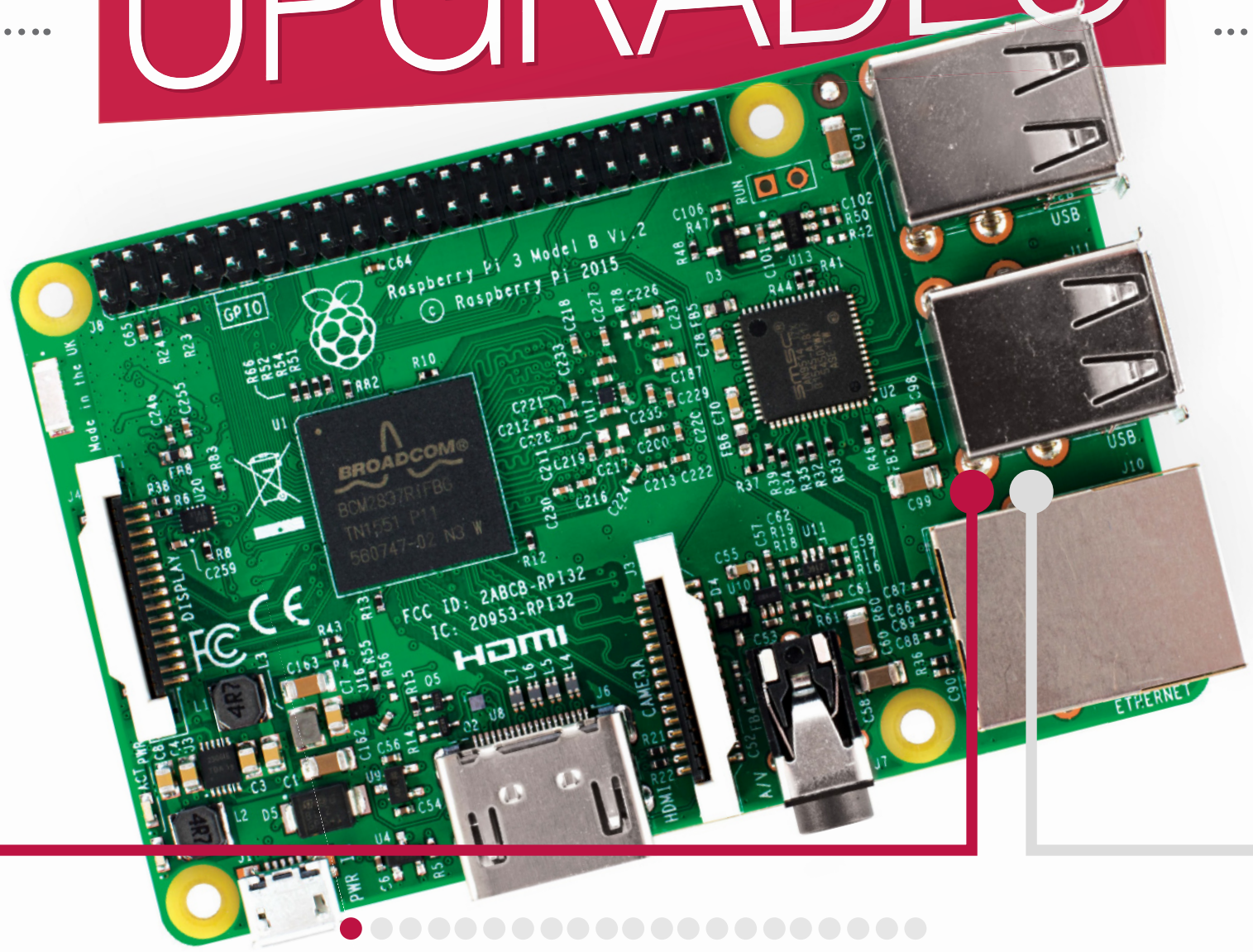
Join the conversation at...

@linuxusermag | Linux User & Developer | RasPi@imagine-publishing.co.uk



Contents

10 AWESOME RASPBERRY PI UPGRADES

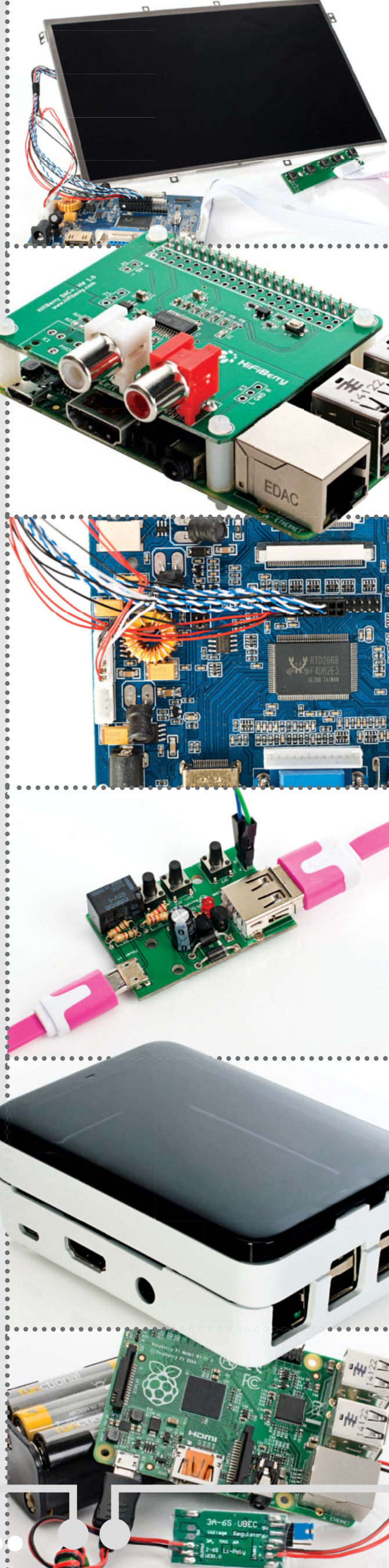


In its short but packed life so far, the Raspberry Pi has been an absolute game changer – not just as a piece of reduced price hardware, but for nurturing a community of like-minded individuals with a common goal: learning and making awesome stuff!

We can't recall the number of times we've browsed over to the Raspberry Pi blog and been blown away by the brilliance of a new project. From sorting Portuguese mail to making bullet time rigs, there are a lot of incredible projects out there – more and more are surfacing every day. People often ask what they can do with a Raspberry Pi and it is actually sometimes difficult to articulate an answer to that question, as the use cases are so broad that it is hard to do the Raspberry Pi justice.

When comparing the Raspberry Pi to your average off-the-shelf computer or mobile device, the brilliance of the Raspberry Pi comes down to its upgradeability and the amount of customisation that is possible. With a smartphone or tablet you can get a trendy case or some cool headphones to go with it, but the customisation with the Raspberry Pi goes far further than that – both in software and hardware. A lot of projects you look at appear to actually be the real-life manifestations of a childhood dream. That ability to turn what used to be dreams into reality is what makes the Raspberry Pi so well loved.

Here we take a look at ten of our favourite Raspberry Pi upgrades, which will help you bring your ideas to life and serve as some inspiration for your next project!



01

Fully protect your Pi

Short Crust Plus

The Raspberry Pi is a durable and reliable little computer, especially when you consider that it is just a populated circuit board with no real protection. However, there may be times where you want to give your Pi a nice shell. Maybe because you want your Pi-based home theatre to look more sleek next to all of your other electronics, or maybe you just want to keep the dust off your tiny computer when carrying it around in your pocket.

DETAILS

Price
£8.99 / $15.95

Available from:
http://bit.ly/1ICXbvw

Left The Short Crust Plus keeps your Pi safe

Left The Pi fits snugly inside the Short Crust Plus case

The Short Crust Plus is one of our favourite cases for the Model B+ and 2B Raspberry Pis due to its sleek, tidy design and well thought-out features. It is also easy to use – the Pi itself snaps into place inside the case and the lid also clicks into place. Each case comes with a set of self-adhesive rubber feet and a free extension that enables you to increase the height of the case in order to accept any add-on boards you might be using.

"The Short Crust Plus is one of our favourite cases for the Model B+ and 2B Raspberry Pis"

02

Portable & solar powered

PiJuice

You can now get hold of an elegant little add-on board that lets you take your projects off-grid and away from mains power sources. PiJuice is compliant with the Raspberry Pi HAT (Hardware Attached on Top) specification and makes use of a slim, off-the-shelf mobile phone battery, and some intelligent charging and power circuitry, to make your Pi truly portable. There's also a version called PiJuice Solar that enables solar recharging and is even capable of taking inputs from other renewable energy sources.

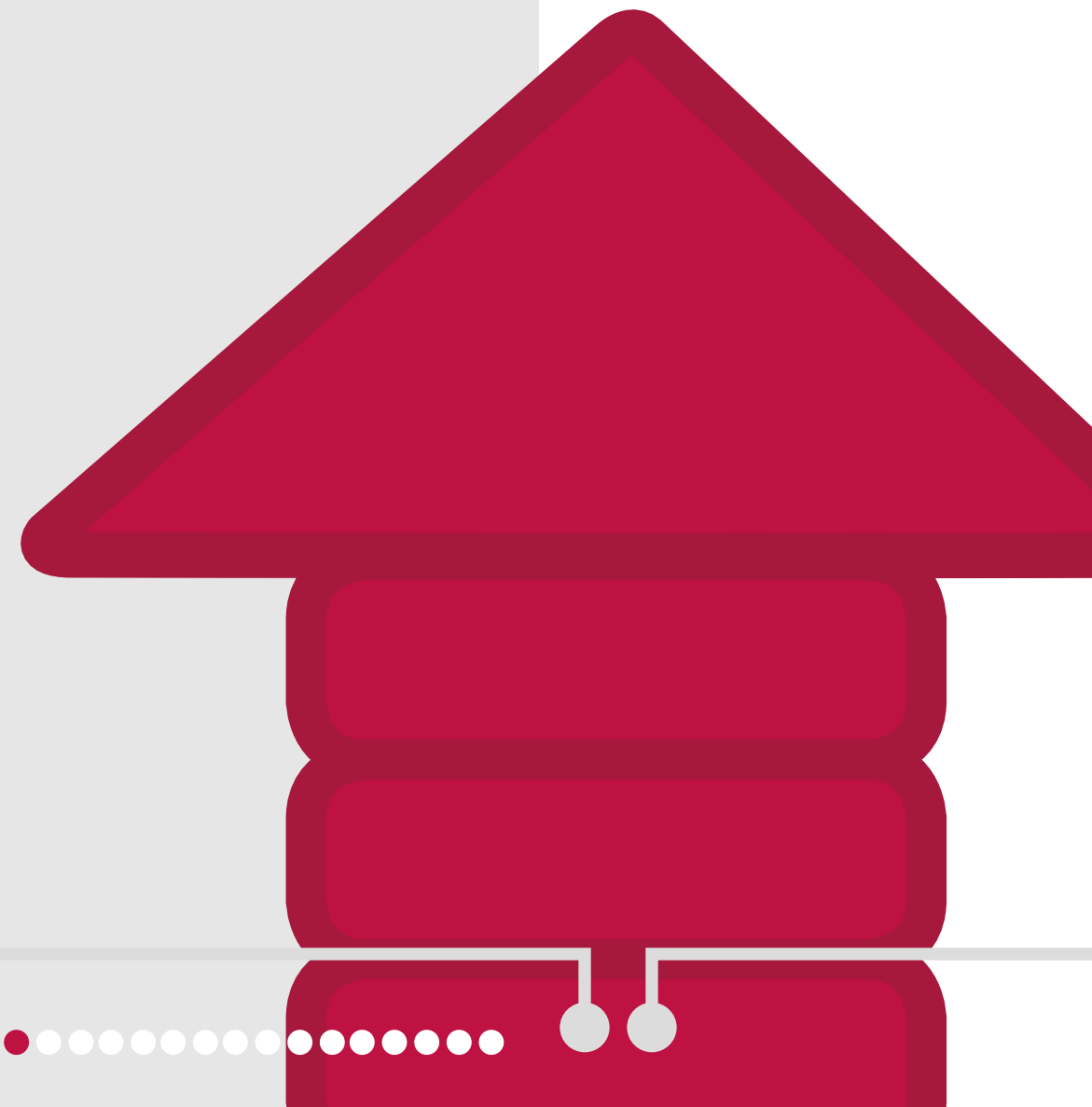
DETAILS

Price
£25 / $39

Available from:
http://bit.ly/1Fb1ywy

Left Get off the grid with this add-on power pack

PiJuice also has a powerful ARM Cortex M0 processor that provides deep sleep functionality, a real time clock, watchdog timers and plenty of other very useful features. The firmware and GUI (graphical user interface) that comes with the PiJuice communicate with the common ACPI (Advanced Configuration and Power Interface) battery and power APIs for tight integration with Raspbian. PiJuice only uses a I2C power sand one GPIO pin, so most of the GPIO pin bank is left free for use with other projects. It comes as standard with a stacking header to make it extremely simple to add other HATs or add-on boards on top. PiJuice will enable you to make a variety of awesome projects – check out the PiJuice Instructables page: **http://bit.ly/1e2CoGE**.



03

Power switch & file-safe shutdown

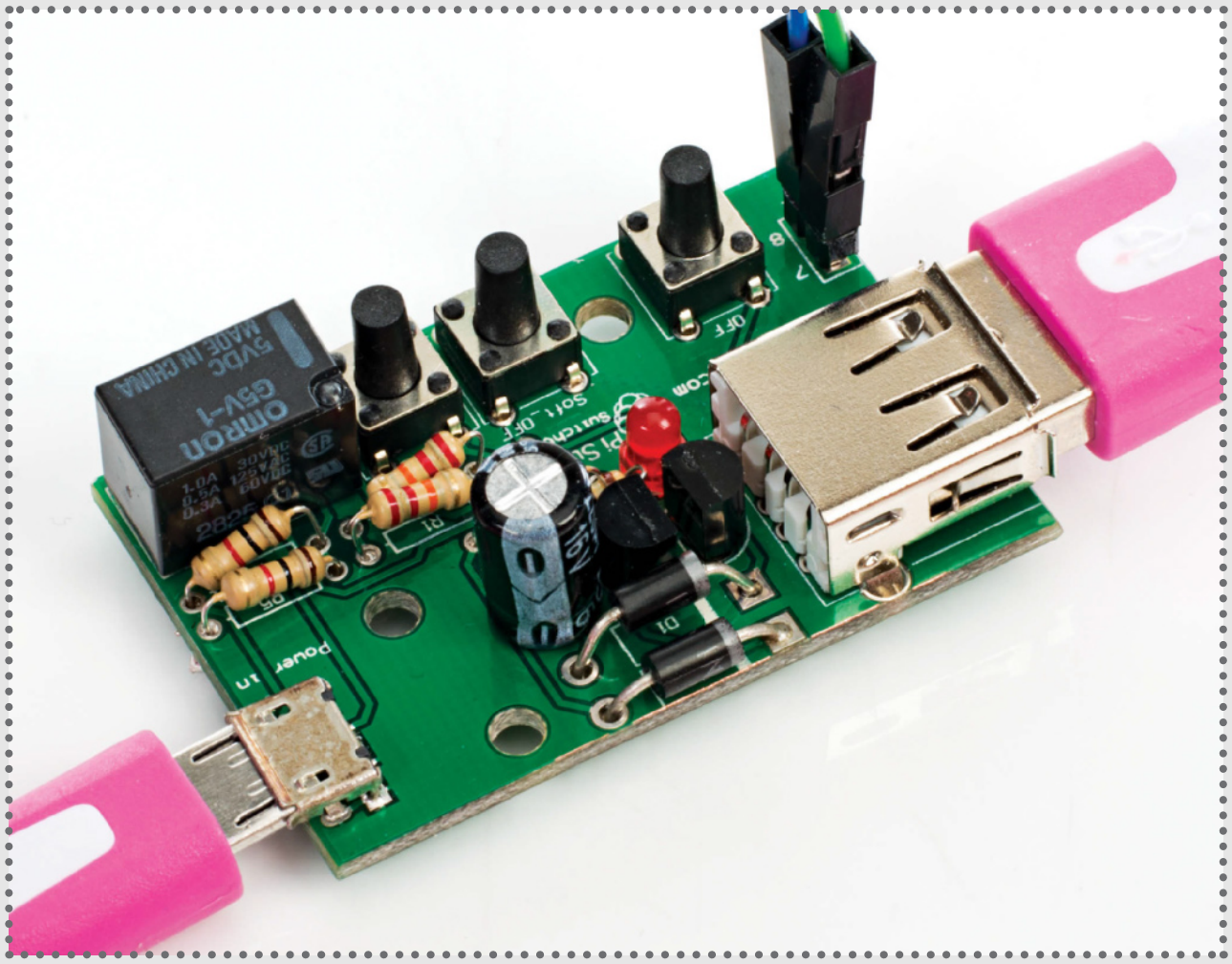
Pi Supply Switch

The Raspberry Pi has been so popular, in part, because of the extremely good value for money of the hardware. It packs a lot of punch for the price point and, because it is designed by a charity, they don't need to inflate the price with high profit margins as much as would be done with a more commercial product. Unfortunately, as with anything low-cost, some compromises had to be made in order to bring it in at such an affordable and small form factor.

DETAILS

Price
£15 / $23.10

Available from:
http://bit.ly/1RXHR0n



Left It's surprising quite how useful a power switch can be

When comparing it to your more standard desktop or laptop computer, one thing that it is obviously lacking is a power switch and power management functionality. It is surprising how something as simple as a power switch can be so very useful, and it is not until you do not have one that you realise this!

The Pi Supply Switch is a self-solder kit which provides an on, off and soft-off (file-safe shutdown) button to give you basic power management functionality for your Pi. With some provided sample scripts you can make sure your Pi is correctly shut down when you switch off – without the need to open any menus or issue any commands in the terminal – and the circuitry in the switch ensures that power is only removed after the Pi has been shut down. As well as making it more convenient for you, it also reduces the possibility of corruption to your SD card from prematurely pulling the power cable.

04

See in the dark with infrared

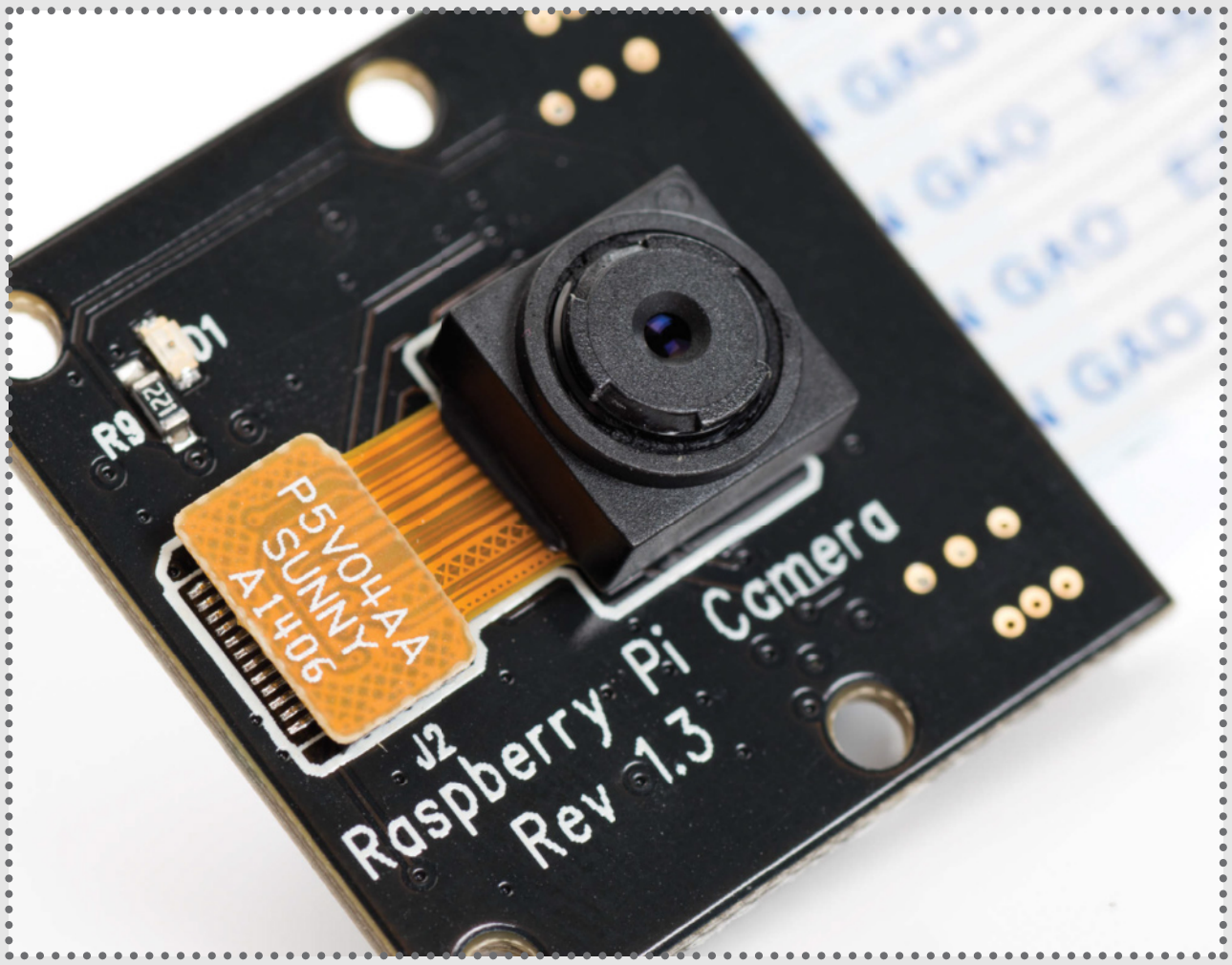
NoIR Infrared Camera

The CSI connector on the Raspberry Pi (between the 3.5 mm jack plug and HDMI connector on the most recent models) enables you to connect a camera module directly without the need for a USB-powered webcam. The camera modules that you can connect here use less power and, as you would expect from the Raspberry Pi Foundation, they come in an impressively small form factor – 25 x 24 x 9 mm, weighing in at around three grams (not including the cable).

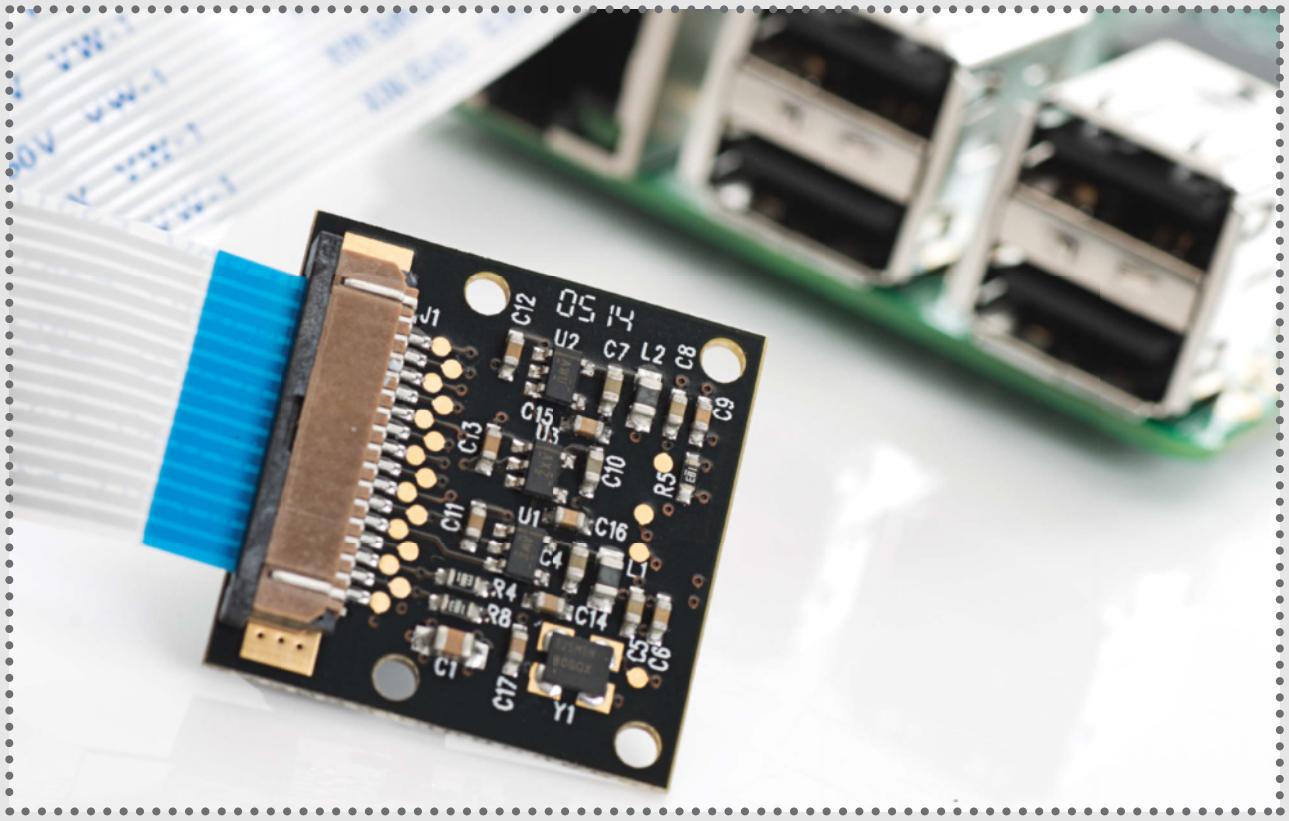
DETAILS

Price
£16.80 / $29.95

Available from:
http://bit.ly/IQyeC4



Left Great for shooting night-time video if you're security-conscious



Left Used in the daytime, the infrared camera can be used to monitor the health of your plants

As you would expect, there is a 'normal' camera module on offer (and by normal, we mean one that captures visible light) with some impressive stats – a 5 MP fixed focus camera, which supports 1080p30, 720p60 and VGA90 video modes (full specs here: **http://bit.ly/1Gy3D8q**). When the camera module was first released, some people clearly were not happy with a visible light camera and had some other (very cool) applications in mind – so they took apart the tiny camera sensor and removed the infrared filter before putting it all back together again. Painstaking work which obviously voids the warranty, but so many people were doing it that the Raspberry Pi Foundation took notice and started doing it themselves, eventually releasing a new infrared camera module – Pi NoIR.

There are some fairly commonplace nighttime uses for infrared video, and if you pair your Pi NoIR with some infrared LEDs (check out the Bright Pi add-on board for this), then you can easily use it for a night vision security camera or a nocturnal animal monitoring setup. Perhaps most amazingly, if you use the infrared camera in the daytime, it can actually be used to monitor the health of green plants (check it out at **http://bit.ly/1QnZdFG**).

05

High quality audio for your Pi

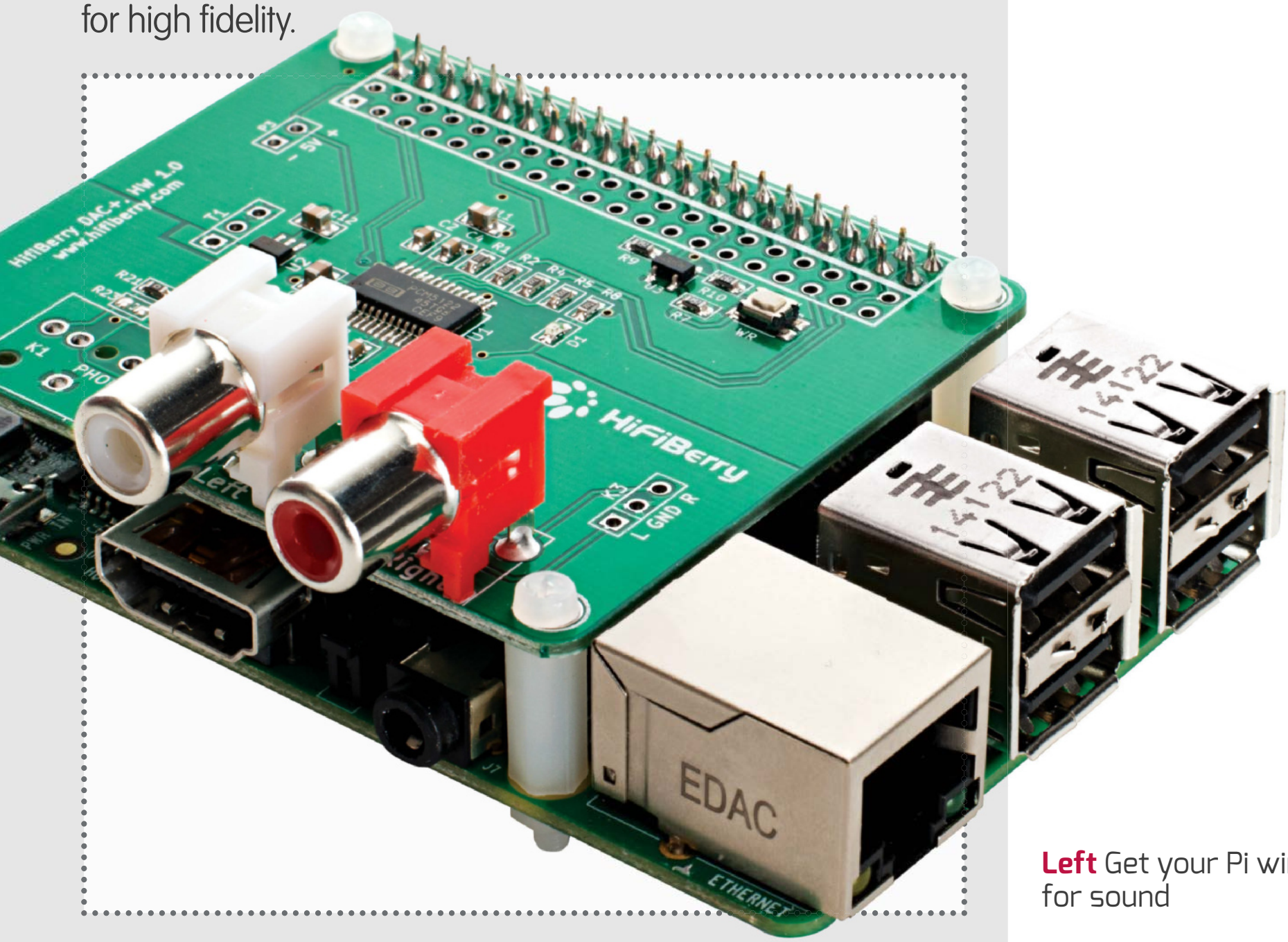
HiFiBerry DAC+

As an educational tool, the Raspberry Pi is pretty much unparalleled due to the support of the very large community that surrounds it. As a high quality audio device, however, you may think it is lacking due to the fact it only has a 3.5 mm stereo output that isn't tuned for high fidelity.

DETAILS

Price
£30 / $34.90

Available from:
http://bit.ly/1L1hh4T



Left Get your Pi wired for sound

Due to its low cost, small footprint and its ability to act as a home media centre, music and video streaming server and much more, you have probably dreamed of enhancing the audio and taking your setup to the next level. The good news is that the clever folk at the Raspberry Pi Foundation, from the second revision of the original Model B, have provided access to the I2S pins; initially on the separate P5 header, and now on the A+, B+ and 2B models it is available from the main 40-pin GPIO header.

I2S is a communications protocol designed specifically for audio devices and has enabled a number of companies like HiFiBerry and IQaudIO to create high quality audio add-ons for the Raspberry Pi. The HiFiBerry DAC+, for example, is an add-on which brings a high resolution (192 kHz, 24-bit) Burr-Brown digital-to-analogue converter to your Pi. It has hardware volume control using Alsamixer, among other features, and as it is a HAT-compatible board. It works plug-and-play out of the box with the latest Raspberry Pi firmwares, and it works with all the popular operating systems for both standard use and media playback, such as Raspbian, Arch Linux, OSMC, OpenELEC, Volumio, Pi MusicBox and many more. If you are serious about your audio quality and want a high quality, low cost, Internet-connected solution, then you no longer have any excuse – you can build your own for under £100!

> “You have probably dreamed of enhancing the audio and taking your Pi setup to the next level”

06

Movement for your camera rig

Pi-Pan Pan Tilt Mechanism

DETAILS

Price
£45.99 / $39.99

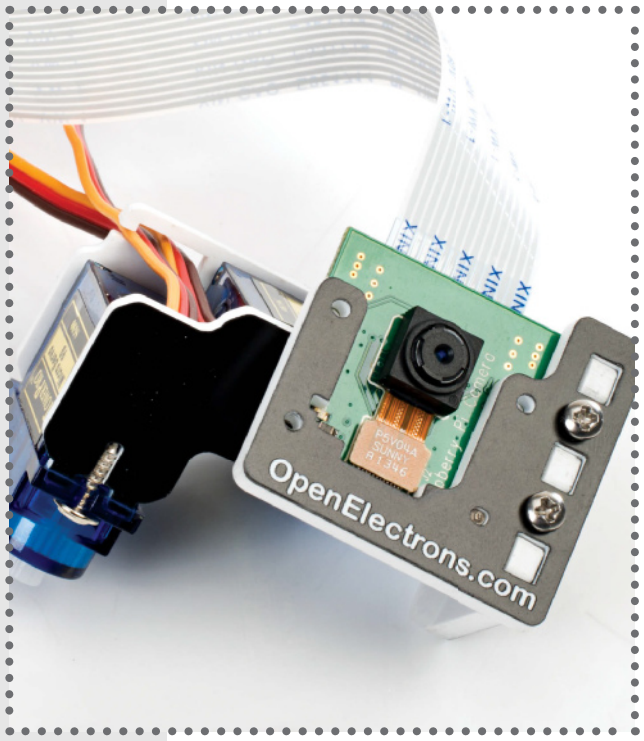
Available from:
http://bit.ly/1dwpEr2

The camera module and Pi NoIR we look at on the opposite page are some pretty essential upgrades to have in your Raspberry Pi toolbox, but what happens if you want to move the camera around to get a different viewpoint? This would be useful in a multitude of projects, such as a robot with a movable camera or an Internet-connected webcam that you can control via a web interface (many IP cameras used for security applications already have a pan-tilt feature, in fact).

The Pi-Pan from Open Electrons is a pan-tilt mechanism for the Raspberry Pi that enables you to articulate the camera by an impressive amount – 110 degrees from top to bottom and 180 degrees from left to right. The kit includes a well considered array of hardware, including a servo driver board, the servo motors required for the actuation and mounting hardware for the camera and servos. On the software side, there are libraries in Python and Scratch so it is easily flexible enough for most projects.

One of the most impressive applications you could use this for is an OpenCV-based motion detection and face-tracking camera. There is sample code available on the openelectrons.com forum and it looks like a truly great project (**http://bit.ly/1JJpXLe**).

Below Pan and tilt the camera by an impressive 110 degrees top to bottom and 180 degrees left to right



07

High definition display & audio

Adafruit 10.1” Display & Audio

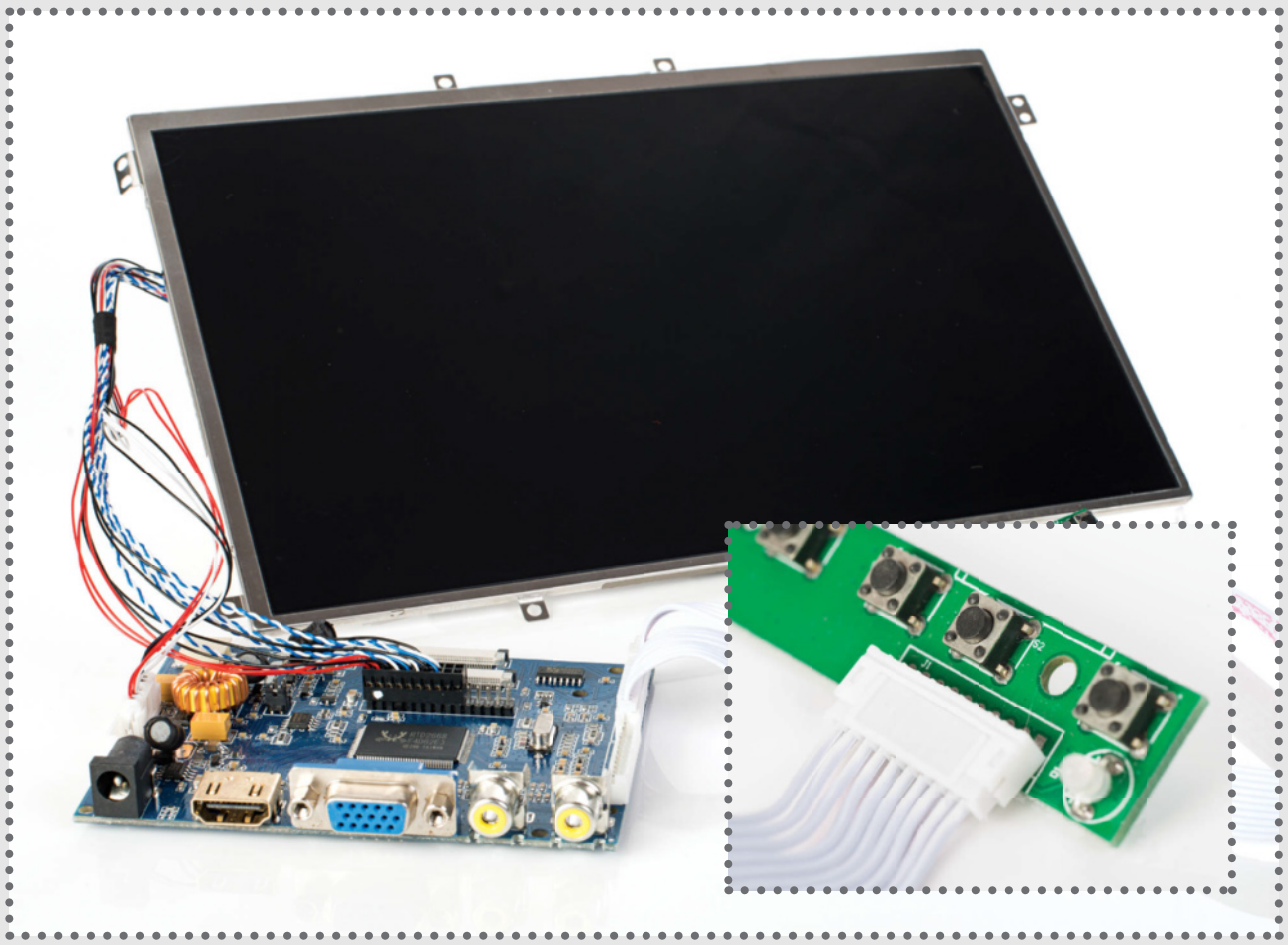
Finding the right display for your project can often be a bit of a pain. We have covered the HDMIPi in a previous issue (**http://bit.ly/1Gb9LNs**), which is a fantastic 9-inch HD screen for your Raspberry Pi, and it was wildly successful on Kickstarter (**http://kck.st/1CuIjwd**).

If you want to take things one step further, Adafruit have a 10.1-inch offering that just can’t be missed. It features a beautiful 1280 x 800 (so slightly higher than 720p) resolution IPS display with a very wide viewing angle. It has mounting tabs to enable you to easily flush-mount it within your project

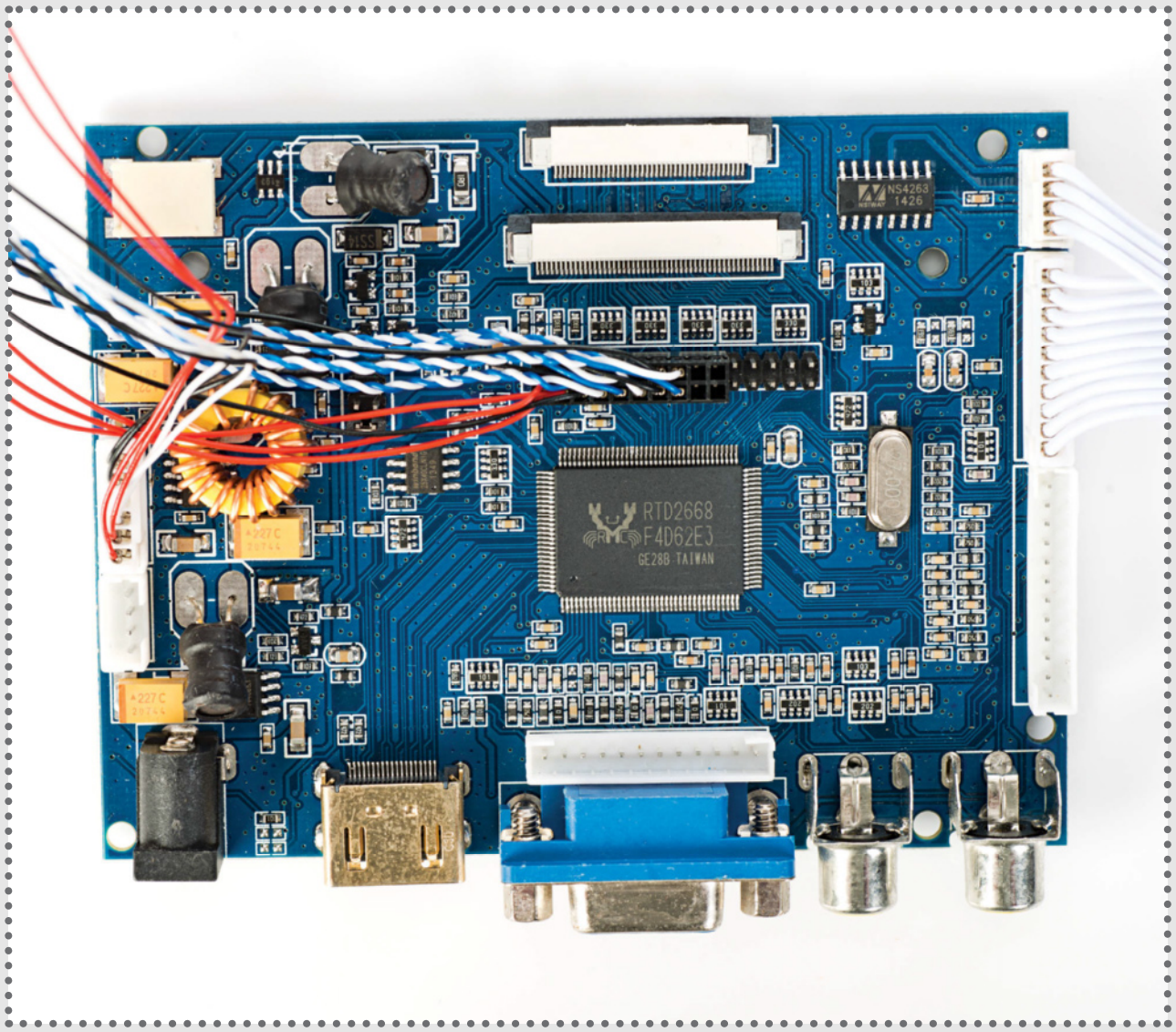
DETAILS

Price
£110 / $154.95

Available from:
http://bit.ly/1HrfR1s



Left Finding the right display for your project can be a pain



Left It accepts a number of different inputs including HDMI, VGA and composite

and it can accept a number of different input methods – HDMI, VGA and composite. Perhaps best of all, this display kit also enables you to directly connect 2-, 4- or 8-Ohm speakers without the need for a separate amplifier or externally powered speaker, which is very useful.

It is not the cheapest display around at $155 on the Adafruit site, but if you need a high quality display in your project with native audio capability then you should seriously consider it. We are already daydreaming of a dedicated multiplayer arcade emulator with built-in stereo audio, and we're sure you can come up with some cool applications too!

"Best of all, this display kit also enables you to directly connect 2-, 4- or 8-Ohm speakers without the need for a separate amplifier"

08

Super low-power displays

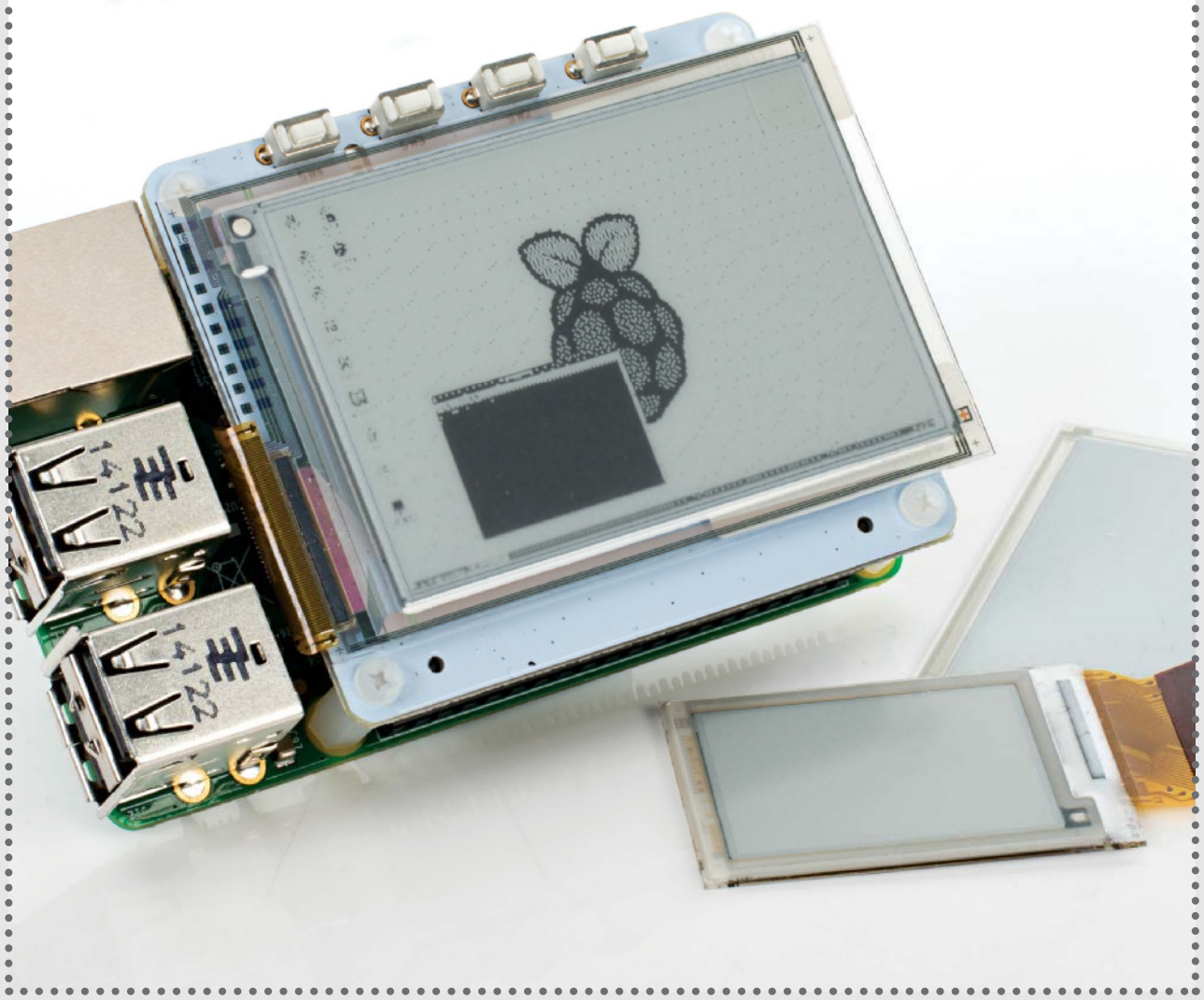
PaPiRus aPaper/eInk HAT

As computers of all sizes and powers are now being embedded into pretty much everything, electronic parts have become even more commoditised and, happily, this is filtering down to display technology as well. We now have a wealth of offerings from your standard monochrome LCDs to TFT, OLED and AMOLED offerings as well.

DETAILS

Price
£30-65 / $47-102

Available from:
http://bit.ly/1f2Lzaj

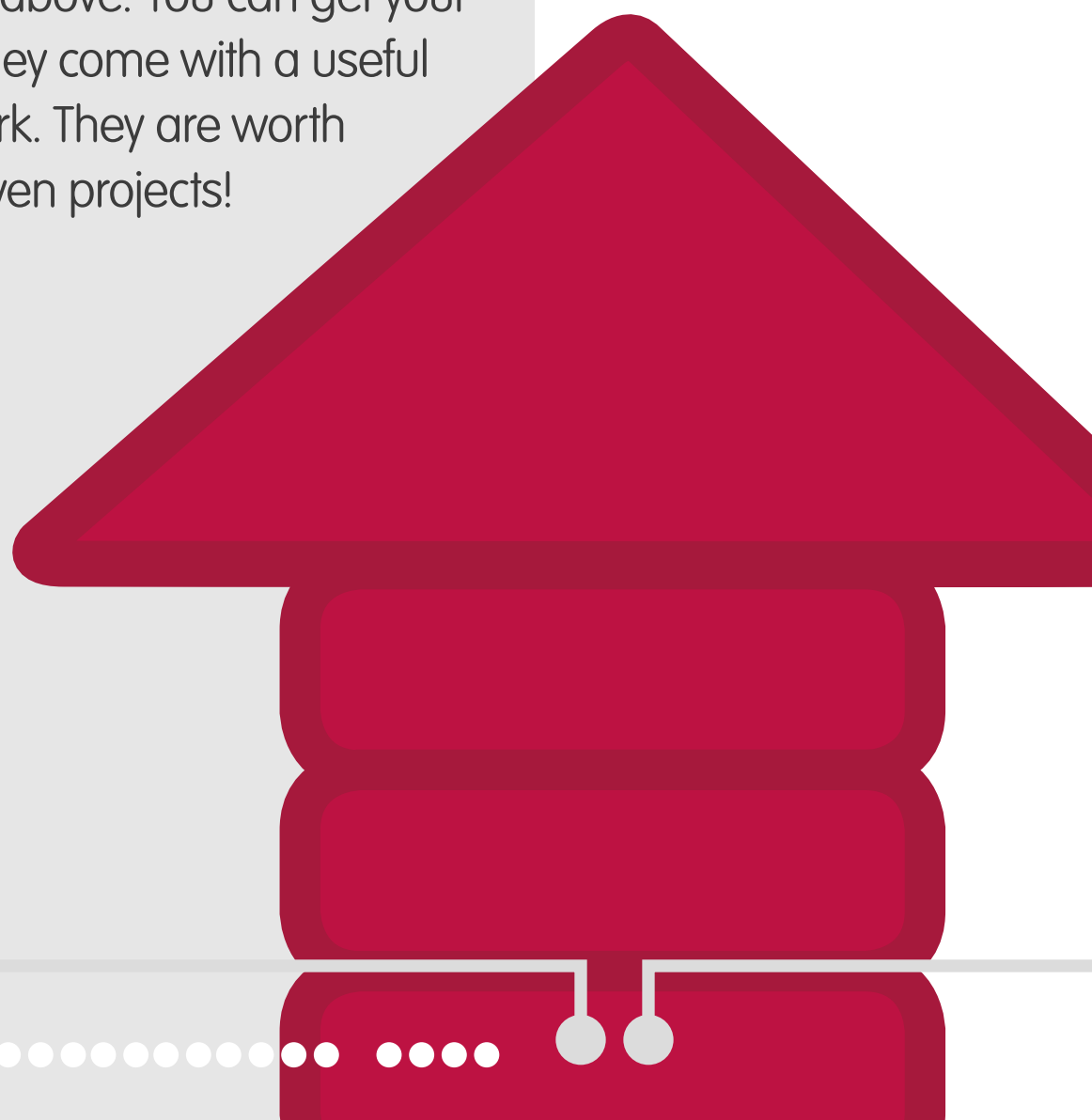


Left ePaper/eInk is one of the most exciting disruptive technologies of recent times

One of the most exciting and disruptive display technologies of recent times is ePaper/eInk. You probably know it best as the screens that go into e-readers like the Kindle and Kobo (fun fact: the Pebble watch is commonly referred to as an ePaper watch, but it actually uses what is known as a Memory LCD and a very clever marketing team). You may have wondered in the past why your iPad barely lasts five hours on a charge but your Kindle lasts for over a week, and the answer is all to do with the display. ePaper only uses power to update what is on the screen, which means that for a large number of applications where you don't need to change screen contents particularly often, it saves a lot of battery power. It would be pretty useless for playing videos, but for e-readers, monochrome graphical info displays, digital price tags, bus and train station signage and many more applications, it is by far the best choice.

PaPiRus brings the low power ePaper display technology you know and love to the Raspberry Pi in a HAT-compatible format with screen sizes ranging from 1.44 to 2.7 inches. The ePaper film used in these screens is actually identical to that in the popular e-readers mentioned above. You can get your hands on one for around £35 and they come with a useful Python and command line framework. They are worth trying out if you have any display-driven projects!

"ePaper only uses power to update what is on the screen, which means it saves a lot of battery power"



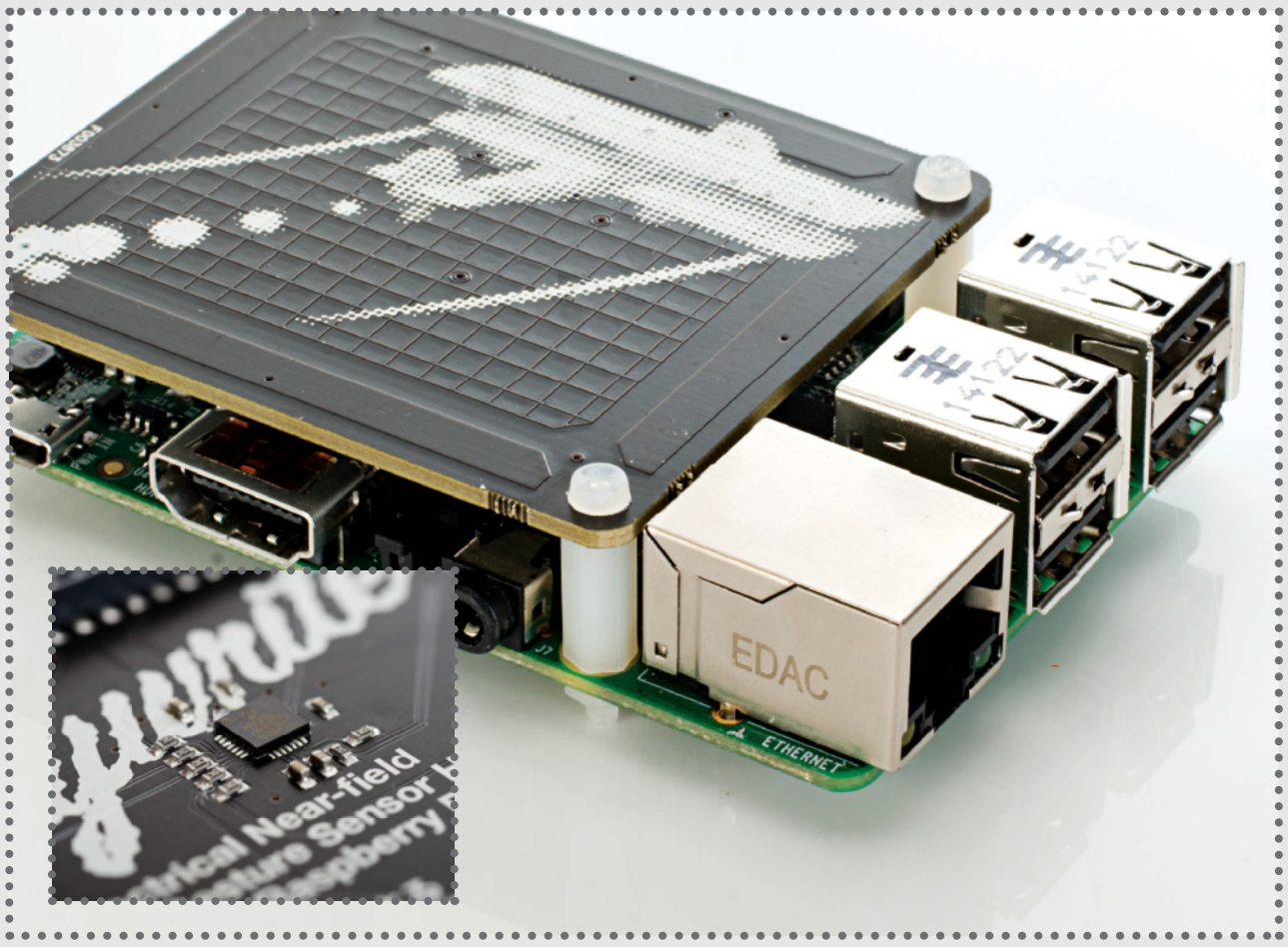
09

Gesture & touch controls

Pimoroni Skywriter HAT

For a lot of projects you undertake with the Raspberry Pi, you will want some kind of user interaction. When using the desktop GUI this is normally done with a keyboard and mouse, but these are not always the most intuitive input methods when you aren't using a full desktop environment and when you don't need to type anything.

The pirates over at Pimoroni have created a new HAT module called the Skywriter that enables you to add near-field 3D gesture and touch sensing to your projects for a



Left Add near-field 3D gesture and touch sensing to your Pi projects

DETAILS

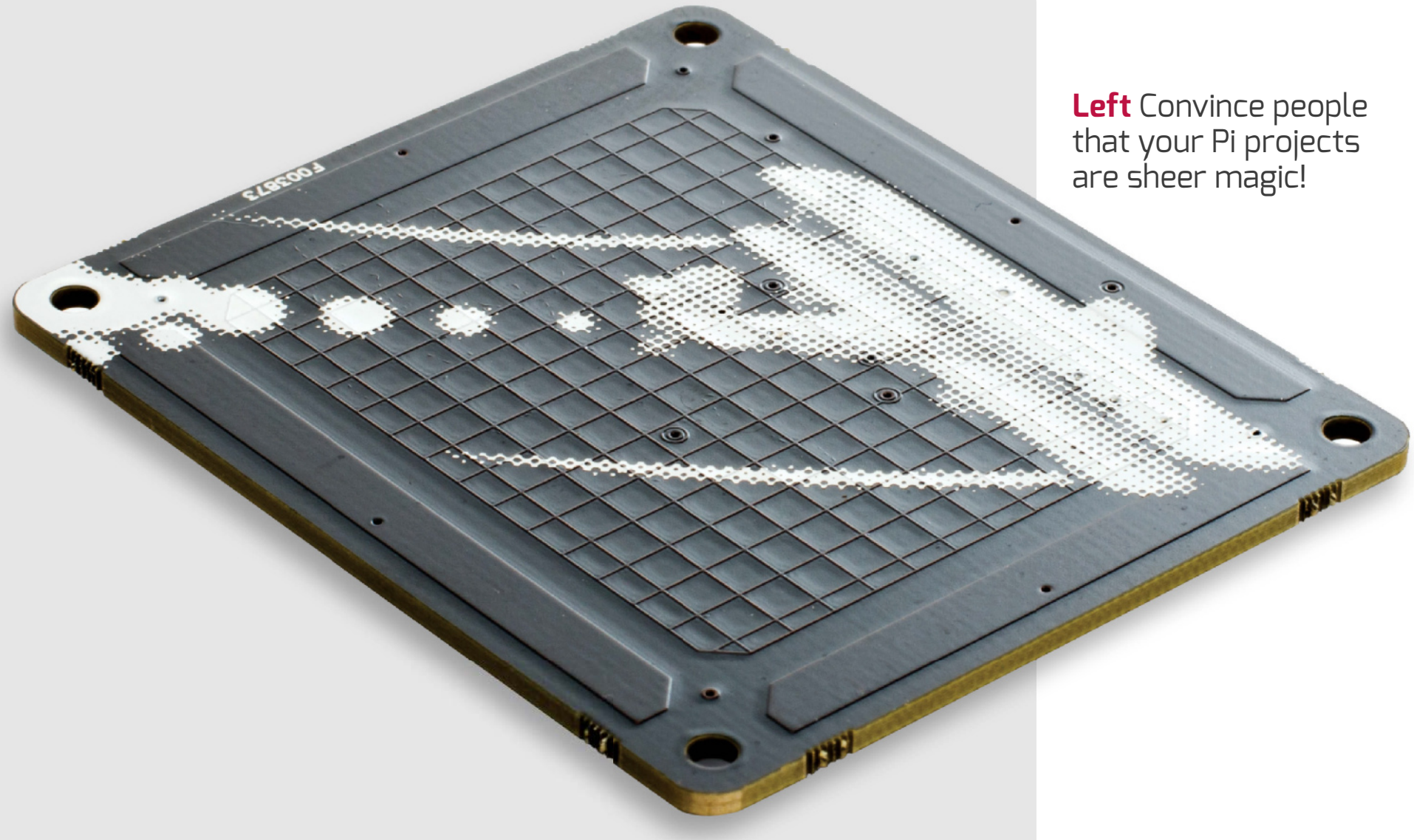
Price
£16 / $20.95

Available from:
http://bit.ly/1IFt9cg

great price. There is a Python API provided that provides full 3D position data and gesture information (swipes, taps and so on). Play with this for a short while and you will realise that it is a really nice input method with a lot of potential – Pimoroni even have a video of a home-made Ras Pi-based theremin (**https://vine.co/v/OrUWTdd0Hlg**).

There is even a larger non-HAT version of the Skywriter that is more than twice the size and boasts a sensing distance of around 15 cm, which means that you can mount it inside your projects behind a sheet of acrylic or other non-conductive material and it will still work. This is especially good if you want to convince people that your projects are simply pure magic.

> "Play with this for a short while and you will realise that it is a really nice input method with a lot of potential"



Left Convince people that your Pi projects are sheer magic!

10

Control your plug sockets

Energenie Pi-mote Control Starter Kit

Home automation is all the rage at the moment – perhaps it is because people are inherently lazy or maybe it's just because this tech is extremely fun to play with! Either way it doesn't really matter, as it can make our lives easier and quicker and can automate tasks that would often be boring and monotonous, like fiddling with heating controls and turning off the lights before you go to bed.

DETAILS

Price
£19.99 / $31

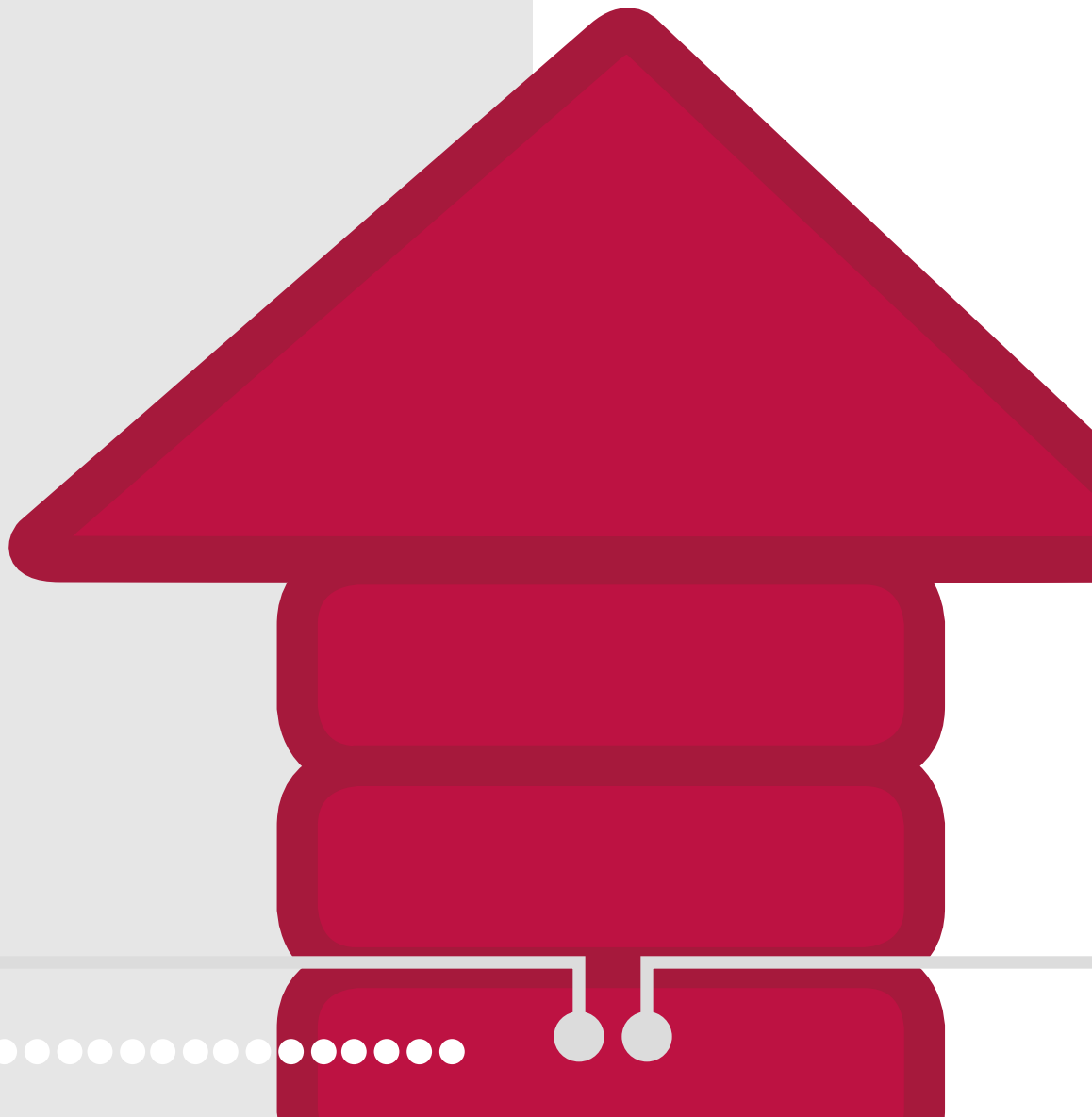
Available from:
http://bit.ly/1L1kYHU

Left Home automation is all the rage at the moment

One thing that we are always told is to turn off devices at the plug rather than leaving them on standby, as they use a lot of electricity when not properly turned off. This is sound advice but is not always a practical solution as the socket is not easily accessible. This is where the Energenie Pi-mote control starter kit comes in. It contains two remote-controlled plug sockets which can be turned on and off with an RF remote. What does this have to do with the Raspberry Pi? Well you also get an add-on board to enable you to control the sockets via software on the Raspberry Pi, which unleashes whole new possibilities – you could set your lamps to turn on and off automatically at specified times when you are away to avoid burglars, or create a basic web app to control your plug sockets remotely using your smartphone.

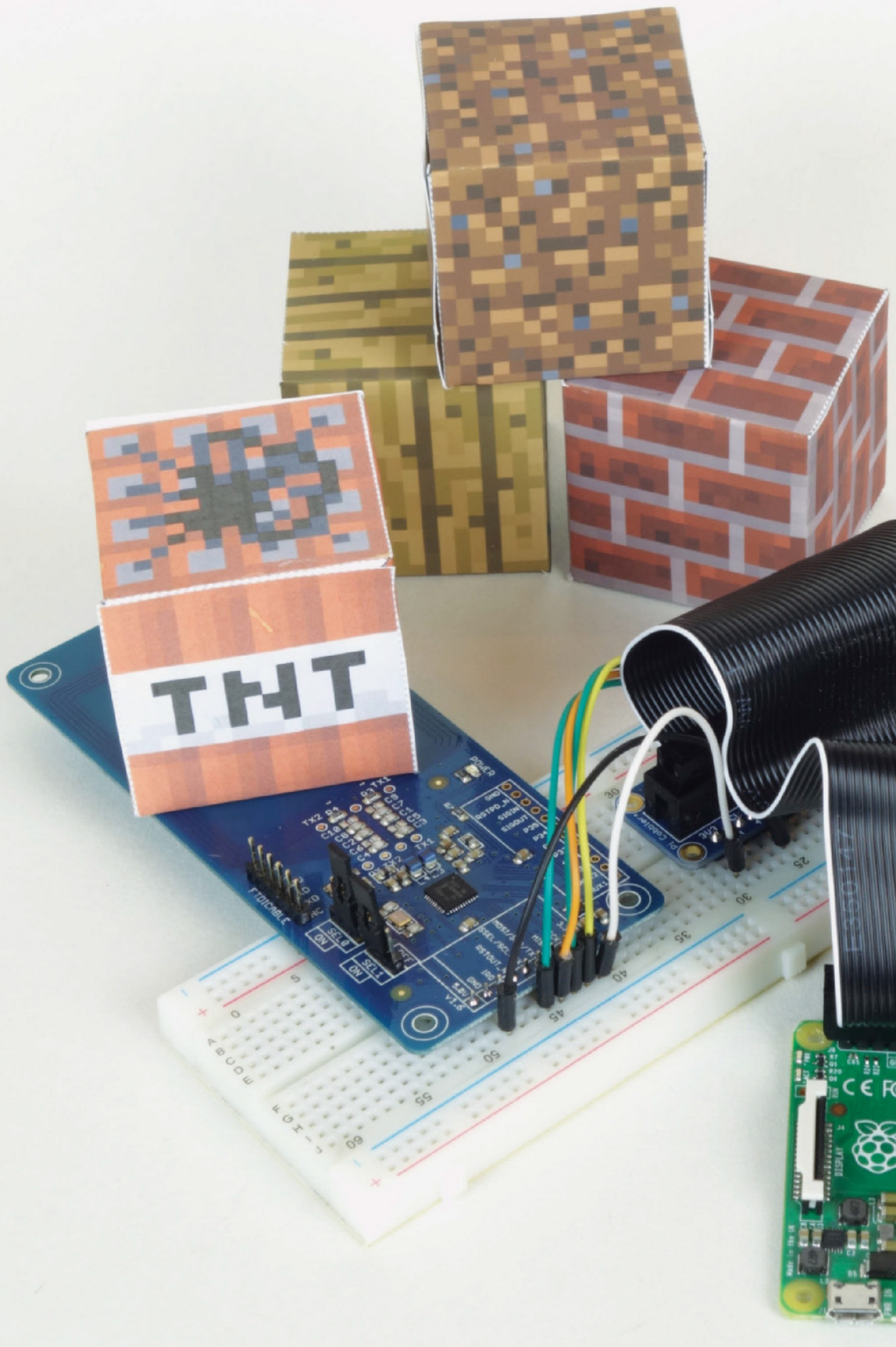
They only come in UK and EU plug types, so if you use a different plug then you may need to look for something else (and maybe send Energenie a request to make more versions).

"You also get an add-on board to enable you to control the sockets via software on the Raspberry Pi, which unleashes whole new possibilities"



Minecraft NFC

Inpired by Amiibos, Tony DiCola makes NFC-enabled papercraft blocks that can be used to build inside Minecraft

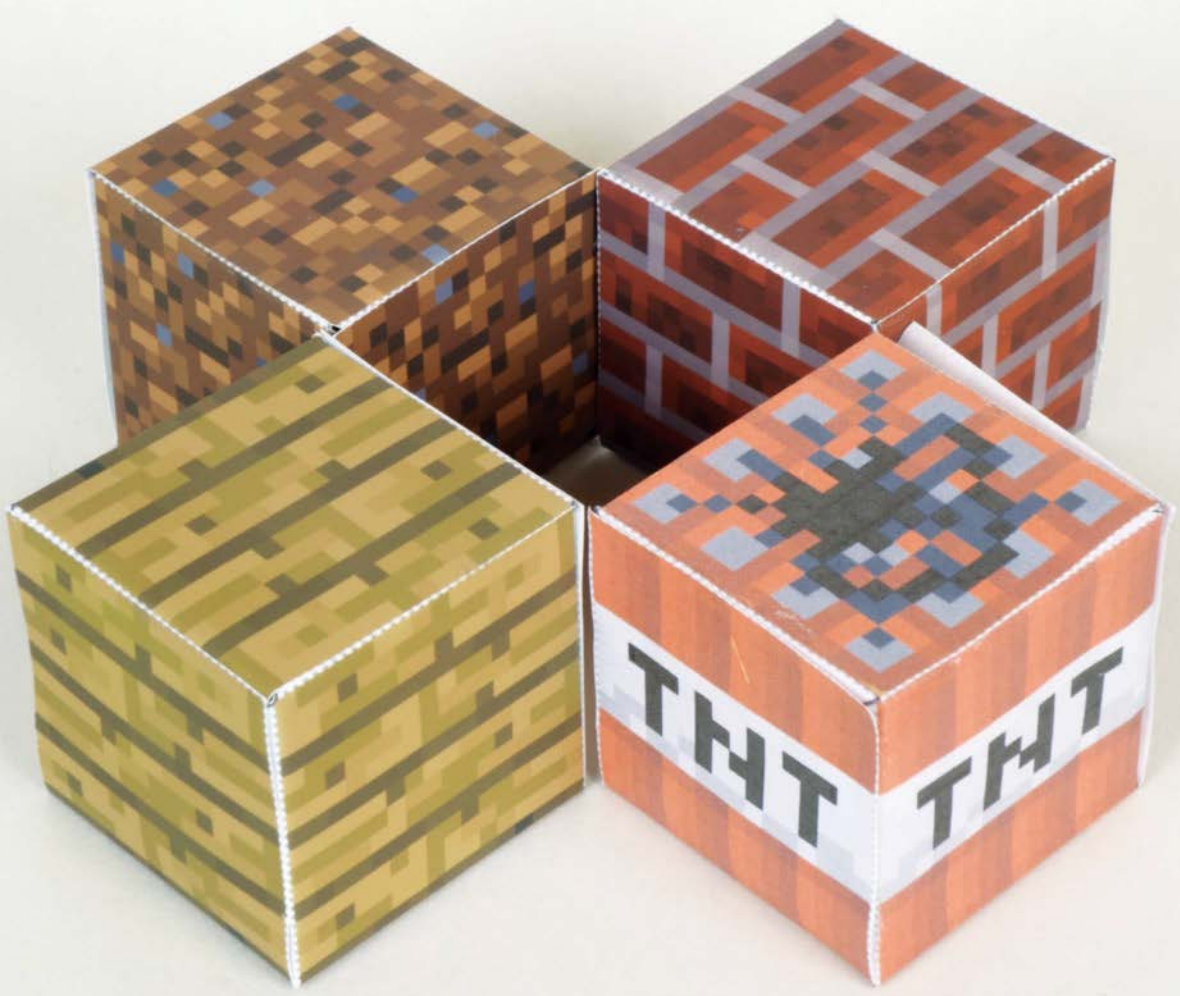


THE PROJECT ESSENTIALS

- Raspberry Pi 2
- Raspberry Pi GPIO Breakout
- PN532 RFID/NFC Breakout
- Breadboard
- MIFARE Classic 1K cards
- Printed and folded Minecraft cubes

Why did you decide to start working with NFC?

I think it was a neat way to explore using physical things with a game – taking a device or an object and making it more interactable. I was a little bit inspired by Amiibos and Skylanders – those little figures you can buy that have NFC in them. So I was thinking, what can I do to make something myself out of this? What's a DIY thing to try? And then I realised that the Raspberry Pi has Minecraft and the cool thing about Minecraft Pi is that it has got a whole little Python API, so it's really easy to use. With a few lines of Python you can create blocks and mess around in the Minecraft world, so I thought that was perfect – I could just put both of these together and make some sort of physical thing to interact with the Minecraft world.

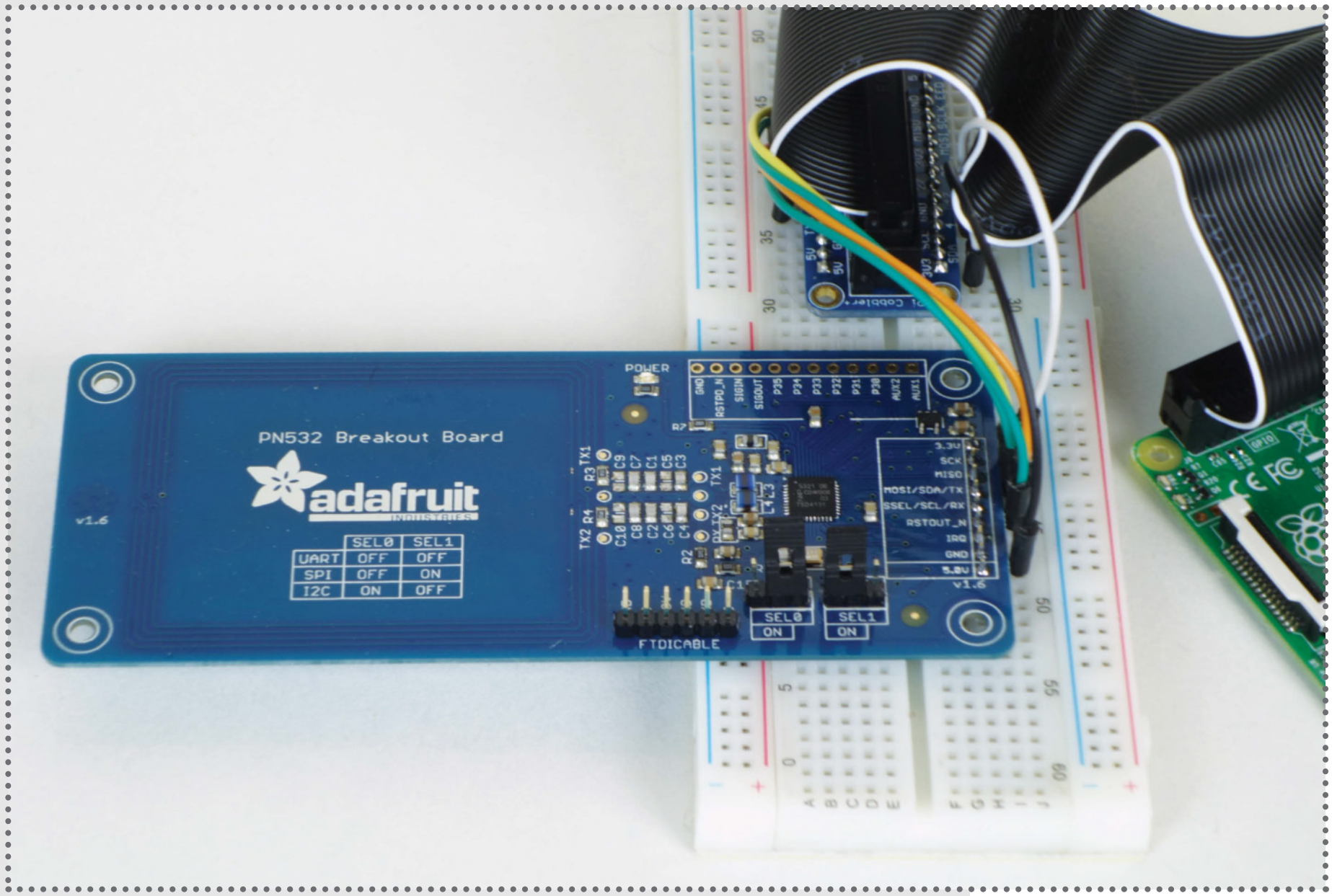


"The cool thing about Minecraft Pi is that it has got a whole little Python API, so it's really easy to use"

So how exactly does it work?

Basically, it has a little NFC reader, which is this circuit board that connects to the Raspberry Pi and uses an SPI connection (a typical connection for embedded devices), and then the Pi has the GPIO ports, so you can get really good low-level access to devices with that. NFC is a wireless Near Field Communication protocol, so it's kind of cool in that you can have these little passive devices, like a tag or a card, and you just hold them near the antenna on the RFID reader and it'll energise them, sending a few little bits of data between them. So the NFC reader is connected to the Pi and there's some Python code that I wrote – a little library – to interact with the NFC reader. There's a little program that you run (also in Python) that basically talks to the NFC reader, and it's in a passive listening mode, so it

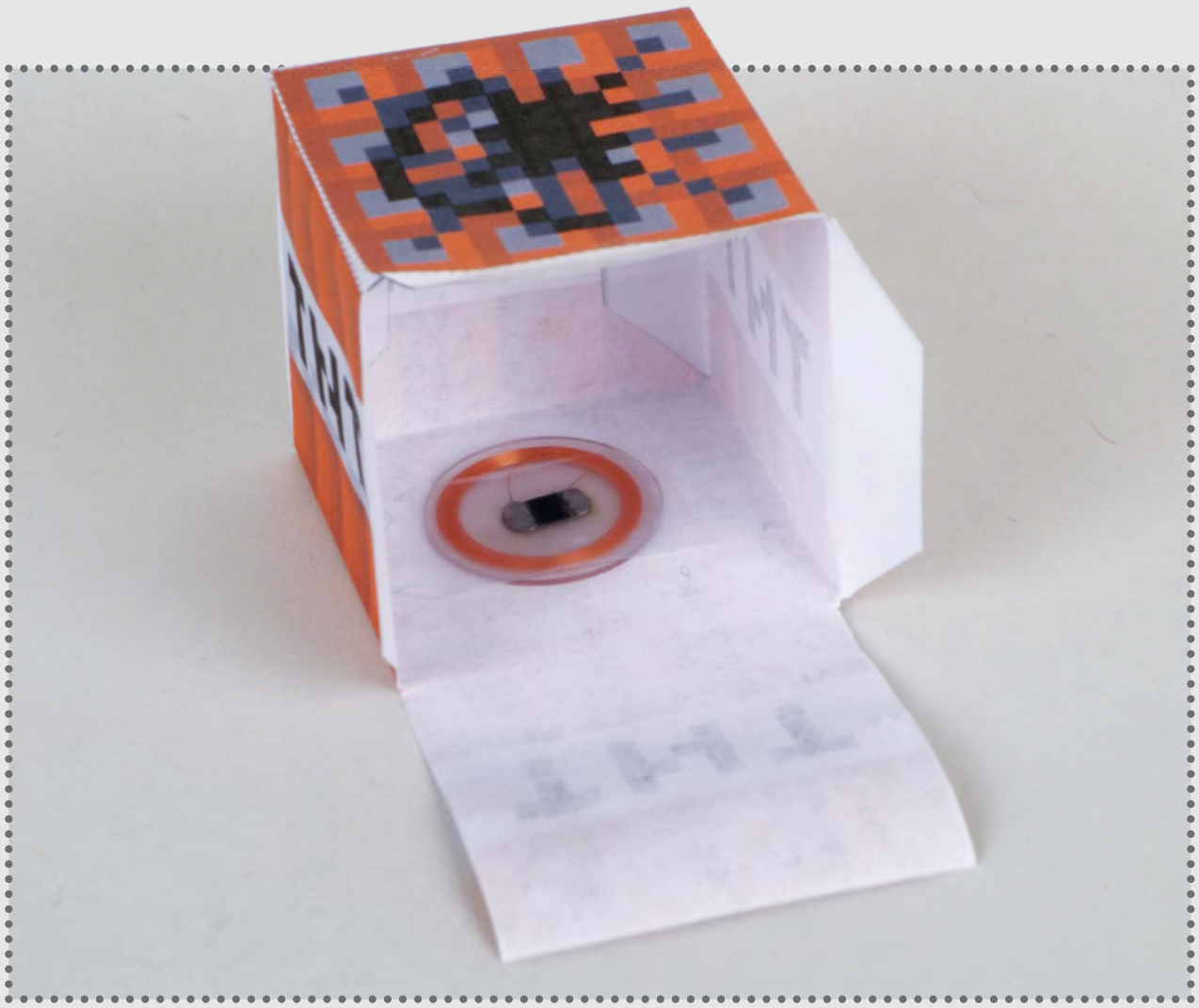
Below The NFC reader connects to the Pi thanks to an SPI connection



just waits to see when a card is swiped. Ahead of time, you build these little blocks and put NFC cards inside them and then you scan those – each one has a unique ID associated with it, so once you get the ID then you can have a little configuration that says 'this ID equals the wood block', 'this ID equals the TNT block', etc. So the program just waits to see a block that's swiped and then, using the Minecraft API, it tells Minecraft to create a new block wherever the player is standing now.

So you wrote the library for the Adafruit PN532 RFID breakout that you are using in this project?

Exactly – the Python port of it. There's also an Arduino port and it's pretty standard, a typical kind of embedded code. The nice thing with this breakout is that you don't really have to talk at a super-low level with the



Left The papercraft blocks talk to the Pi via NFC

different NFC protocols – the chip on here takes care of that for you, so really the library is just talking to the chip and saying, 'run the listening command' or 'run the read or write command'. It abstracts away how to initialise the device and it has a bit of a framing protocol – when you send a command it has a hash and other stuff associated with it to make sure it gets the correct data, so the library takes care of all that for you. You can just look at the datasheet and it tells you all the different commands that it supports, and in order to send those commands you have to frame them in the right way, but the library will do that for you. And even above that there are some higher-level functions that just, for example, wait for a card, and then once you have a card there's a function that says 'give me all the data from the card', so it's meant to be real simple and easy to use, really make you able to be productive with it.

> "It's meant to be real simple and easy to use, (to) really make you able to be productive"

How much information can you store in these NFC tags? Could you write the layout for a Minecraft building on a chip?

The cards that I was using, the MIFARE Classics, store 1KB of data and I'm not actually even storing much data – it's just the type of block, a byte or two. But you could actually create some arrangement of blocks, like a pyramid or a house, up to a kilobyte, so you'd have some restrictions in that each block will need a position. You'll probably need three or four bytes for that (or maybe even three or four 16- or 32-bit values), so it could probably store maybe 100 blocks or so in an arrangement on a 1KB card. But there are bigger cards – I think there are 4KB ones. So it could be a cool thing to support for the future, extend it to support a structure or something that you make ahead of time – the tricky thing is that you'd need

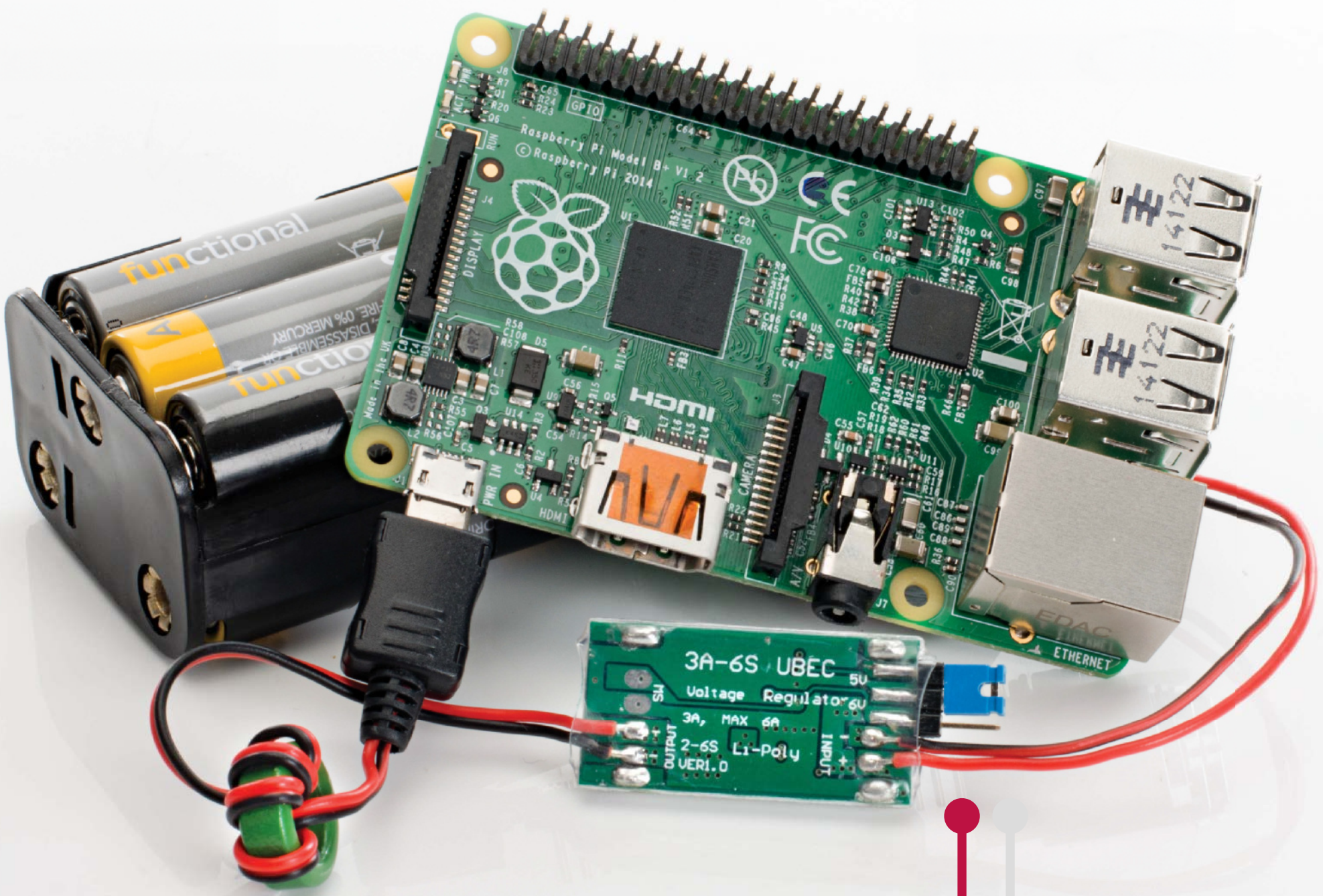
some kind of editor, ideally. Minecraft on the Pi is pretty basic, so you would have a bit of a tough time trying to define the structure without just dropping someone down to configuration and saying, 'Okay, write out the exact position of the blocks'. But I guess if someone were really ambitious, they could make a little 3D editor or something to place the blocks.

Have you experimented using the PN532 in any Raspberry Pi projects outside of Minecraft up to now?

Not too much yet. I was kind of curious to see if we could read Amiibos – what do they actually store on the cards? – and you can actually read them. There's a community on Reddit trying to hack the Amiibos and actually figure out what the data is, but it's all encrypted by Nintendo so you can't really understand it. I was hoping that maybe you could actually clone an Amiibo but the 532 unfortunately doesn't support writing to the Amiibo; they use special tags – NTAG216, I think – and it's a weird format, quite hard to find. Nintendo didn't want people to copy them, obviously!

Add a battery pack to your Raspberry Pi

Don’t leave your Raspberry Pi behind – incorporate it into mobile projects by powering it with AA batteries



Your Raspberry Pi's mobility is usually restricted by the length of the power lead. Rather than limiting it to your desk or living room, however, you can use it for mobile projects as diverse as launching it into near-Earth orbit or monitoring and automating your garden.

Of course, to do this you will need batteries, but adding battery power to your Raspberry Pi is simpler than you might have imagined. All that is required are six rechargeable AA batteries (or single-charge alkaline), a battery box with space for the batteries and a UBEC. The latter is a Universal Battery Elimination Circuit, a voltage regulator that will regulate the power supply and prevent damage to the Raspberry Pi, and can be bought for under £10.

THE PROJECT ESSENTIALS

AA battery box
http://bit.ly/1FDiJGa

3-Amp UBEC
http://bit.ly/1HLKih7

3-Amp terminal strip

6x AA rechargeable batteries

01 Order your components

If you're buying your components online, you should be able to get them all within five days. However, if you're ordering offline (specifically the UBEC), you should avoid traditional electronics stores and instead visit a model enthusiast store, as these circuits are regularly used in RC devices.

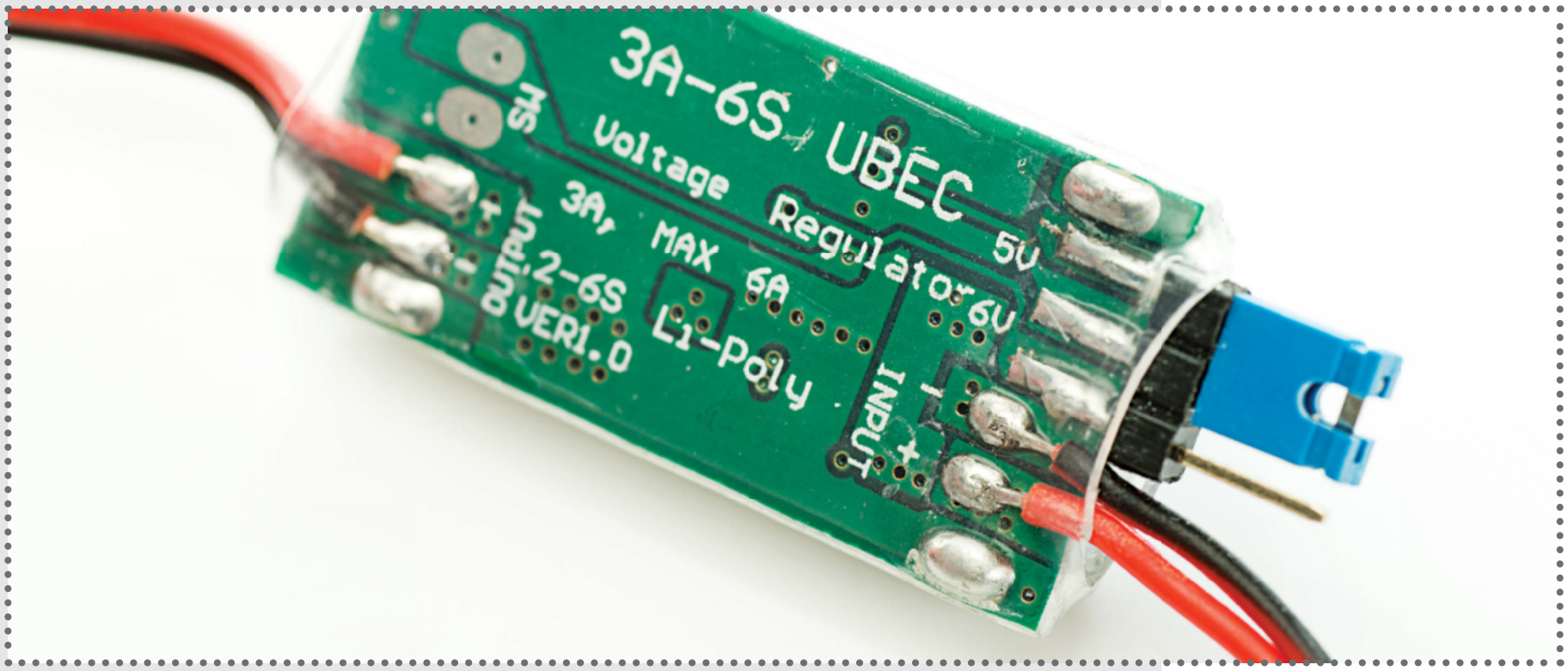
02 Check your UBEC

Two types of UBEC are available. If you used the store that we suggest in the resources box to the right, you'll receive one with a micro USB power connector for easy connection to your Raspberry Pi. However, if you bought one from eBay then there is a strong chance that you will receive one with a 3-pin connector.

03 Change the UBEC connector pins

To use the UBEC with a 3-pin connector, alter the position

"Adding battery power to your Raspberry Pi is simpler than you might have imagined"



of the pins so that they occupy the two outer slots. Use a small jeweller's screwdriver to lever up the small plastic catch and remove the red wire from the central slot, before sliding into the unoccupied outer slot.

Above Alter the position of the pins so that they occupy the two outer slots

04 Connect the UBEC to the battery box

With five batteries in the battery box, connect it to the UBEC: red-to-red, black-to-black. You might do this by twisting the wires or soldering, or employ a 3-amp terminal strip, cut down to two pairs. The terminal strip can be cut to size using a modelling knife.

05 Add a battery to boot

With your Pi ready to use and your Wi-Fi dongle plugged in, connect the UBEC to the micro USB port and insert the sixth battery into the battery box. The Pi's power and status lights should indicate that the computer is booting up, which gives you a fully portable computer.

06 Connect the 3-pin UBEC

If you purchased the UBEC with the now-modified 3-pin

connector, you'll need to connect this to the Raspberry Pi's GPIO header. Specifically, connect the positive +5V (red) connector to Pin 2 and the negative 0V connector to Pin 6. Once again, check the status lights to ensure the Pi is booting.

07 Measure uptime

You should have already set up your Pi for SSH use, so connect to the device via Putty after giving it time to boot fully (at least 60 seconds). In the terminal, enter:

```
watch -n 60 uptime
```

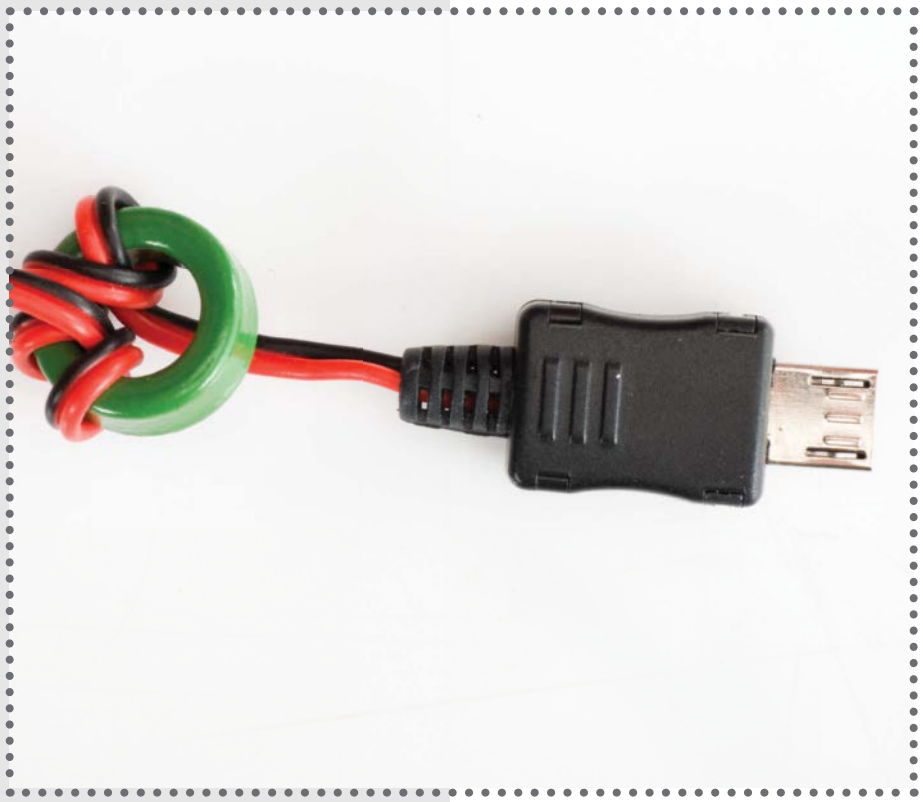
This command will display the system uptime and also keep the Wi-Fi connection active.

08 Judge your uptime results

Uptime results depend upon the type of battery you use and the Raspberry Pi model. Single-charge batteries will last a little bit longer, but this is a more expensive option. Meanwhile, newer models have greater power requirements but run for less time. For more power, add more batteries!

09 Power extreme!

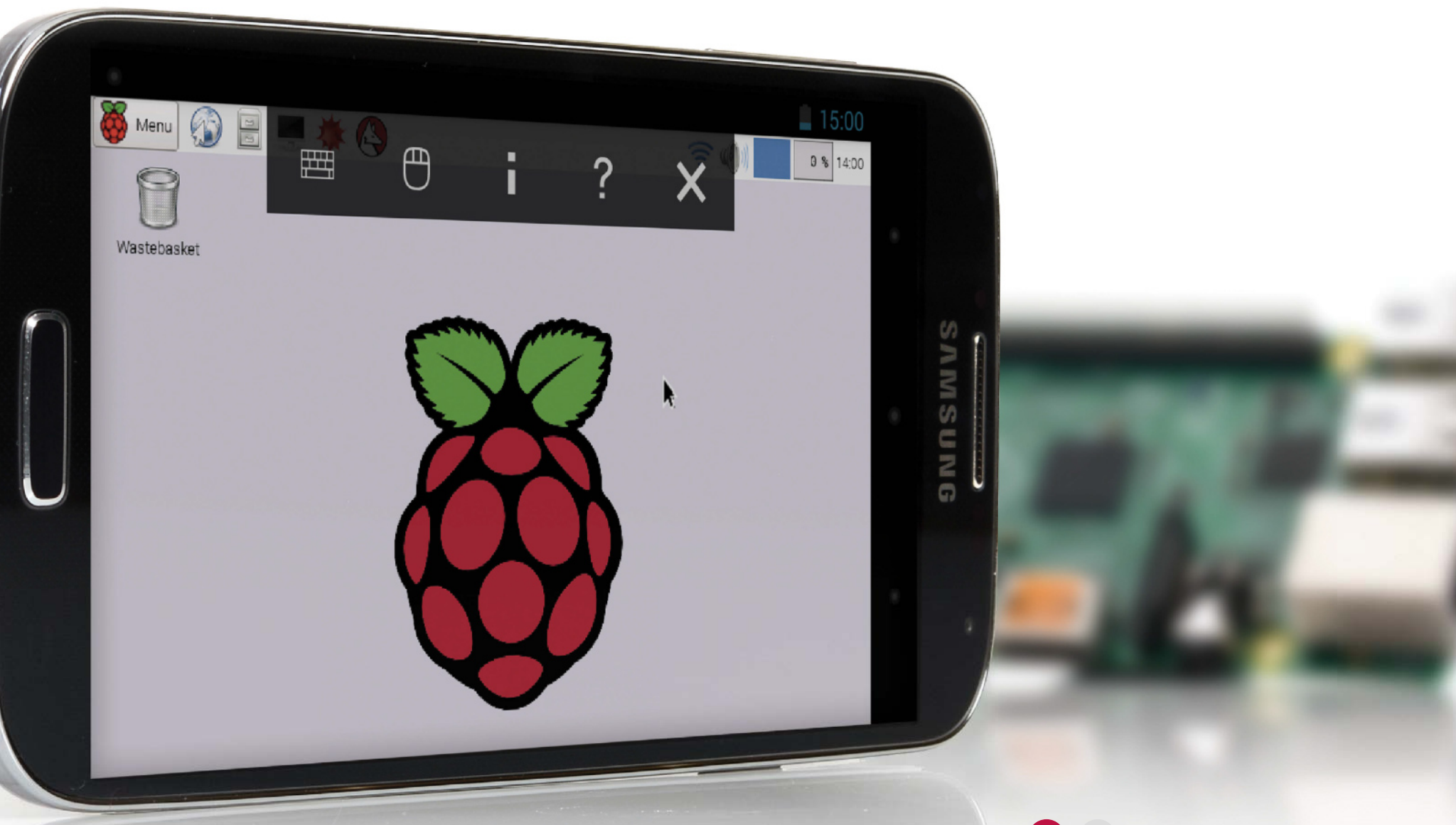
More batteries added in parallel should result in almost double the uptime (at least 16 hours on a 256MB Raspberry Pi Model A), but instead of alkaline or rechargeable batteries you might consider a modern lithium-based AA cell, which will last considerably longer than alkaline batteries.



Below You can also use a UBEC to charge your smartphone from a battery pack

Use an Android device as a Raspberry Pi screen

Connect to a Pi with your phone or tablet using VNC as a secondary or actual display



There are a few ways to attach a display to a Raspberry Pi. The ones that everyone is most familiar with are the HDMI port, which can go straight into your monitor or TV, or via the GPIO ports with a portable screen. One avenue that is rarely pursued is using VNC software to remotely view the Raspberry Pi desktop using another computer entirely.

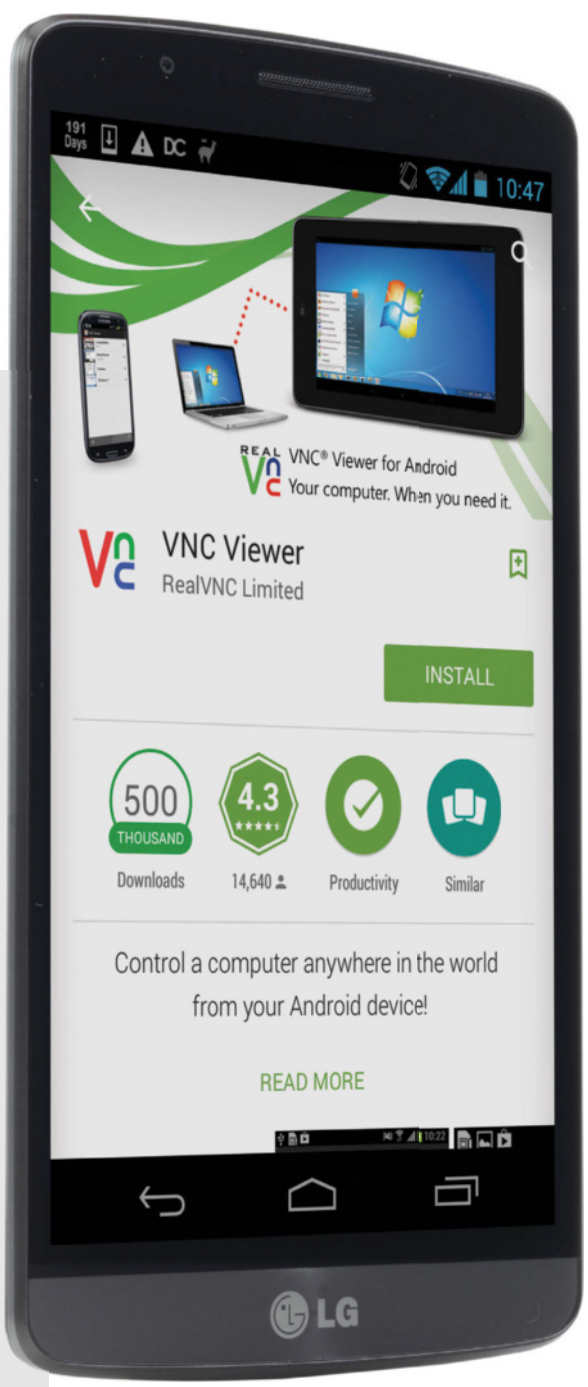
It's actually fairly simple and we are going to concentrate on viewing your Pi screen via Android for maximum portability. With further tweaks you will be able to use this on the go if you find yourself commuting and wanting to catch up on your favourite program, for example.

There are a few things that you should be aware of when doing this project – it can be quite taxing on the Pi and is likely to drain a bit of battery from your phone too. We will talk about some options to help take the strain off the Pi if the connection seems a little laggy and how this method can be used on normal PCs as well, so you can use a range of devices and get the most from your Raspberry Pi.

THE PROJECT ESSENTIALS

TightVNC
http://tightvnc.com

VNC Viewer
http://bit.ly/1jCzBRJ



01 Update your Raspberry Pi

Before starting, you should absolutely make sure that your Raspberry Pi is up to date. With the limited resources on the Pi, any optimisations can make the experience better. You'll first want to update the software by opening the terminal and using:

```
$ sudo apt-get update && sudo apt-get upgrade
```

… and then follow it up with a firmware upgrade by running:

```
$ sudo rpi-update
```

02 Install the software

We're going to be using TightVNC for our VNC needs

Above An Android phone screen makes the perfect display

here – specifically, the server side of the software. We'll need to install it first though, so head back to the terminal and type:

```
$ sudo apt-get install tightvncserver
```

You may have to run through a setting or two as it installs, hit 'yes' (or 'y') to them to continue.

03 Start up VNC

Once everything's installed we can actually start up the VNC server – it's probably a good idea to do so now and check to make sure it's all working. We'll do a test run with the following:

```
$ vncserver :1 -geometry 640x480 -depth 16
-pixelformat rgb565
```

As long as everything is installed correctly, it should start without any problems at all.

04 First time setup

The first time you turn it on, you will have to set a password. You'll need to pick a password you can remember as it will also be used by the client as it connects via VNC. You can only use a maximum of

"Look up the resolution of your Android device and then modify the line used to start up the server"

eight characters for the password though, so think carefully about how you want to make it secure.

05 Stop and restart

As TightVNC doesn't quite work like a normal service, you can't do a usual service X restart (or equivalent) to turn it off or whatever you wish to do with it. Instead, if you want to restart it, you'll have to manually turn it off and then restart it with the original command. To kill it, use:

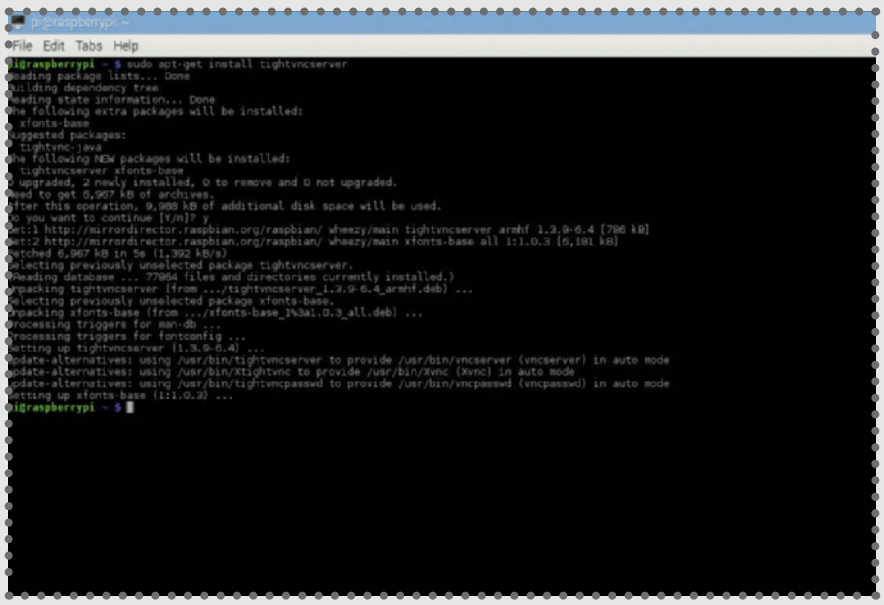
```
$ vncserver -kill :1
```

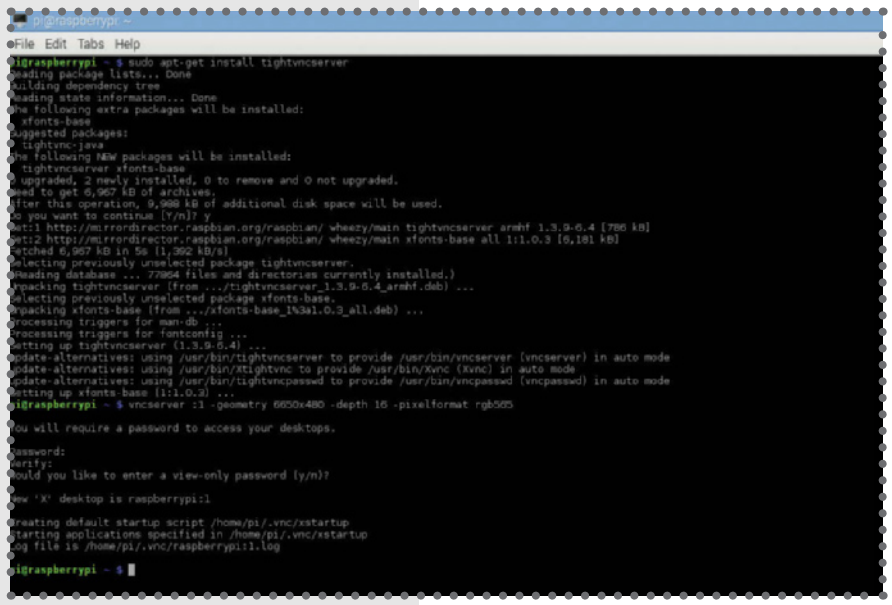
… replacing 1 with the display number that you originally create.

06 Correct resolution

We created our test server with a resolution of 640 x 480, just to get it running. However, this is unlikely to be the resolution of your phone. Luckily, this resolution is not fixed every time, so the best thing to do is to look up the resolution of your Android device and then modify the line used to start up the server. For example, if you have a 1080p tablet, you would use:

```
$ vncserver :1 -geometry 1920x1080 -depth 16
-pixelformat rgb565
```

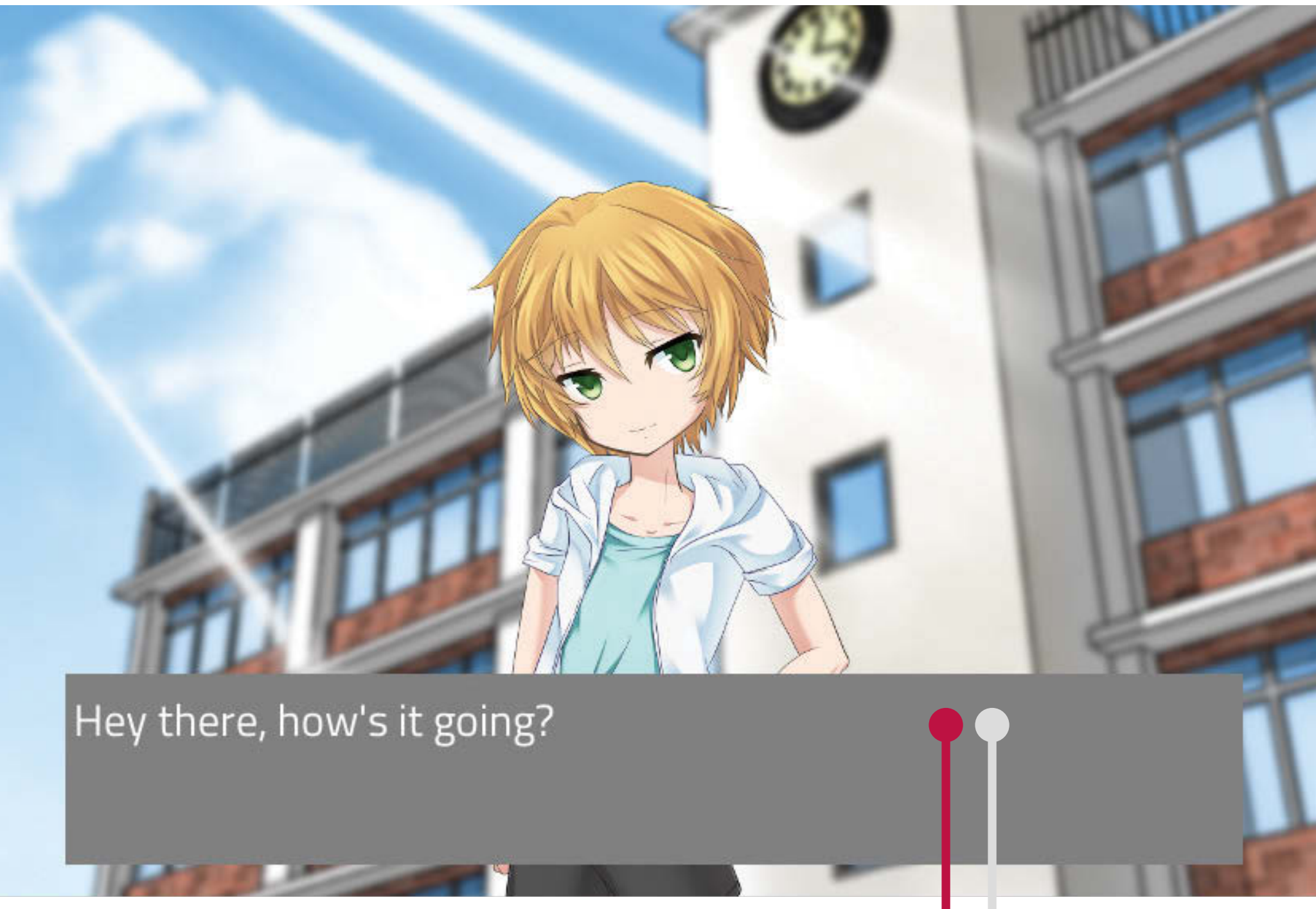




Below Look up the resolution of your Android device and then modify the line used to start up the server

Make a visual novel with Python

Bridge the gap between books and videogames by creating an interactive novel or choose-your-own-adventure with Python and pygame



Videogames have come a significant way since their early days, and in recent years it's become even easier to tell a gripping and compelling story through the medium. A great way to tell a pure story is through the genre of visual novels. These interactive novels are extremely popular in Japan, though they're gaining traction in the rest of the world, and usually have the player click through a story and make decisions as they go to experience different plot points and endings.

In Python, this is a relatively easy project to create, but with the addition of the pygame module we can make it easier still and more expandable for the future. Pygame adds better support for positioning images and text, creating display windows, using mouse and keyboard inputs, and simplifying the coding process. We'll be coding this in Python 2, so make sure to run it in IDLE 2 and not IDLE 3 while writing and testing and coding.

THE PROJECT ESSENTIALS

Python 2
www.python.org

Pygame
http://pygame.org/download.shtml

IDLE Python IDE

Game assets

"These interactive novels are extremely popular in Japan, though they're gaining traction in the rest of the world"

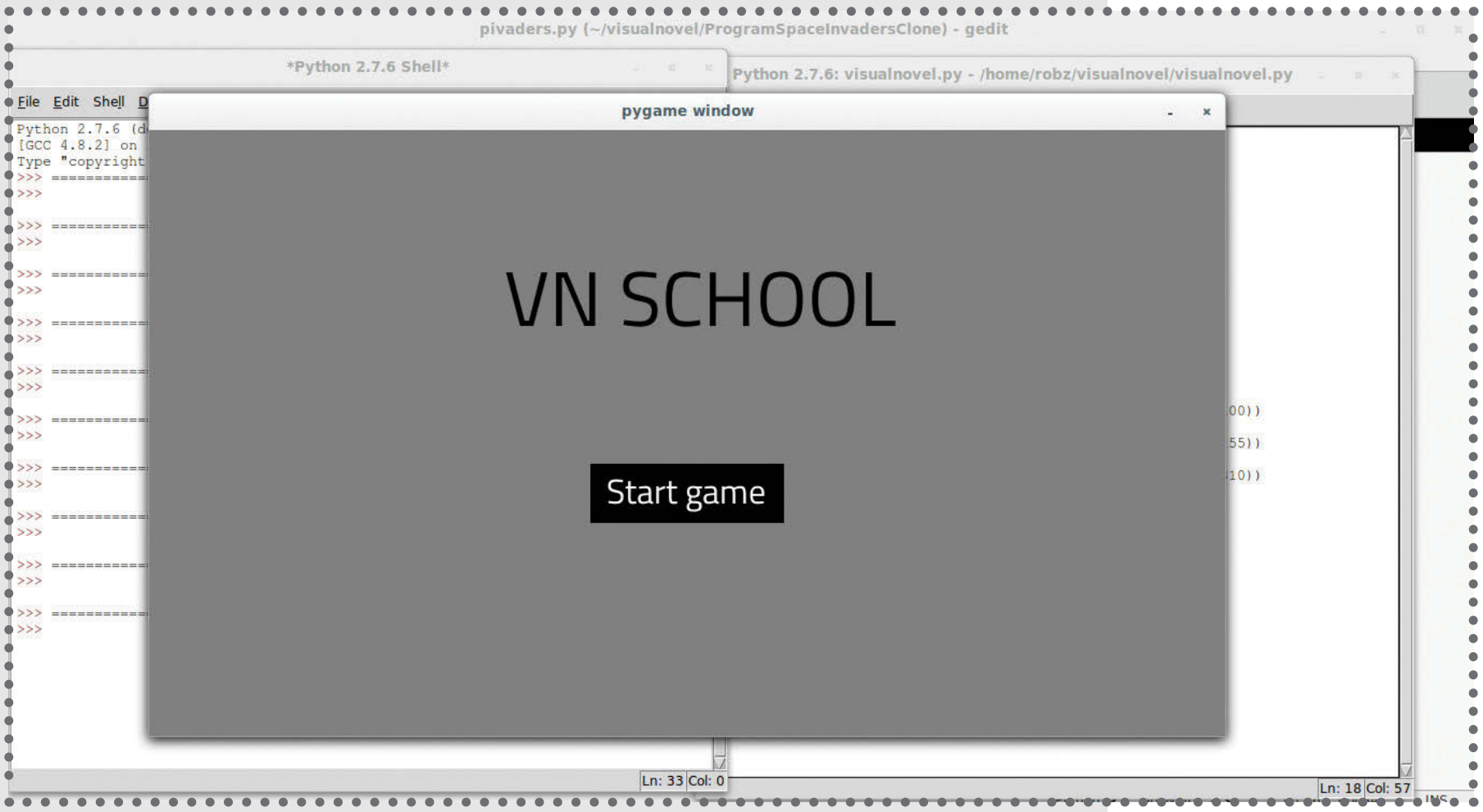
01 Get pygame dependencies

The best way to install pygame for your system is to compile it. To do this you need to first install the right dependencies. Open up the terminal and install the following packages, which in Ubuntu looks like:

```
$ sudo apt-get install mercurial python-dev
python-numpy libav-tools libsdl-image1.2-dev
libsdl-mixer1.2-dev libsdl-ttf2.0-dev libsmpeg-
dev libsdl1.2-dev libportmidi-dev libswscale-
dev libavformat-dev libavcodec-dev
```

02 Get the pygame code

Next we need to download the code for pygame direct



Above Cut out custom templates to suit your project's style and build requirements

from the source. Still in the terminal, you can do this by typing in:

```
$ hg clone  https://bitbucket.org/pygame/
pygame
```

… which will download it to the folder pygame. Move to it using cd pygame in the terminal so we can continue building it.

03 Build the pygame module

To install it, we need to do it in two steps. First we need to prepare the code to install using the terminal with:

```
$ python setup.py build
```

One that's finished, you can then actually install it with:

```
$ sudo python setup.py install
```

This won't take too long.

04 Install in other ways

If the above steps don't work for you (or if it seems a little bit daunting) you can check the pygame website

for binary and executable files that will work on other operating systems and Linux distros. Head to **http://pygame.org/download.shtml** to get the files you need for your specific system, including Windows and OS X. The rest of the tutorial will work in any OS.

05 Get the visual novel files

We've uploaded the code to our website, and here we're going to walk you through what we've done to make it work. Download the files for the visual novel and unzip them. The two files we care about for the moment are the visualnovel.py and script.py python files – this is where all the important code is.

06 Understand the script file

For the moment the script file is small and literally just holds the script for the game. It's made up of events for the visual novel to move between, line by line, by splitting it up into scenes. This includes the location of each line, the character, the actual line itself and information on how the game flows. These are all matrices that store the relevant information, and are completely customisable.

07 How the script relates

In our game, the code pulls in elements from the script

file as it goes. We'll explain how that works later, but this also allows us to implement decisions later on to change where the game might take you in a later part of the game.

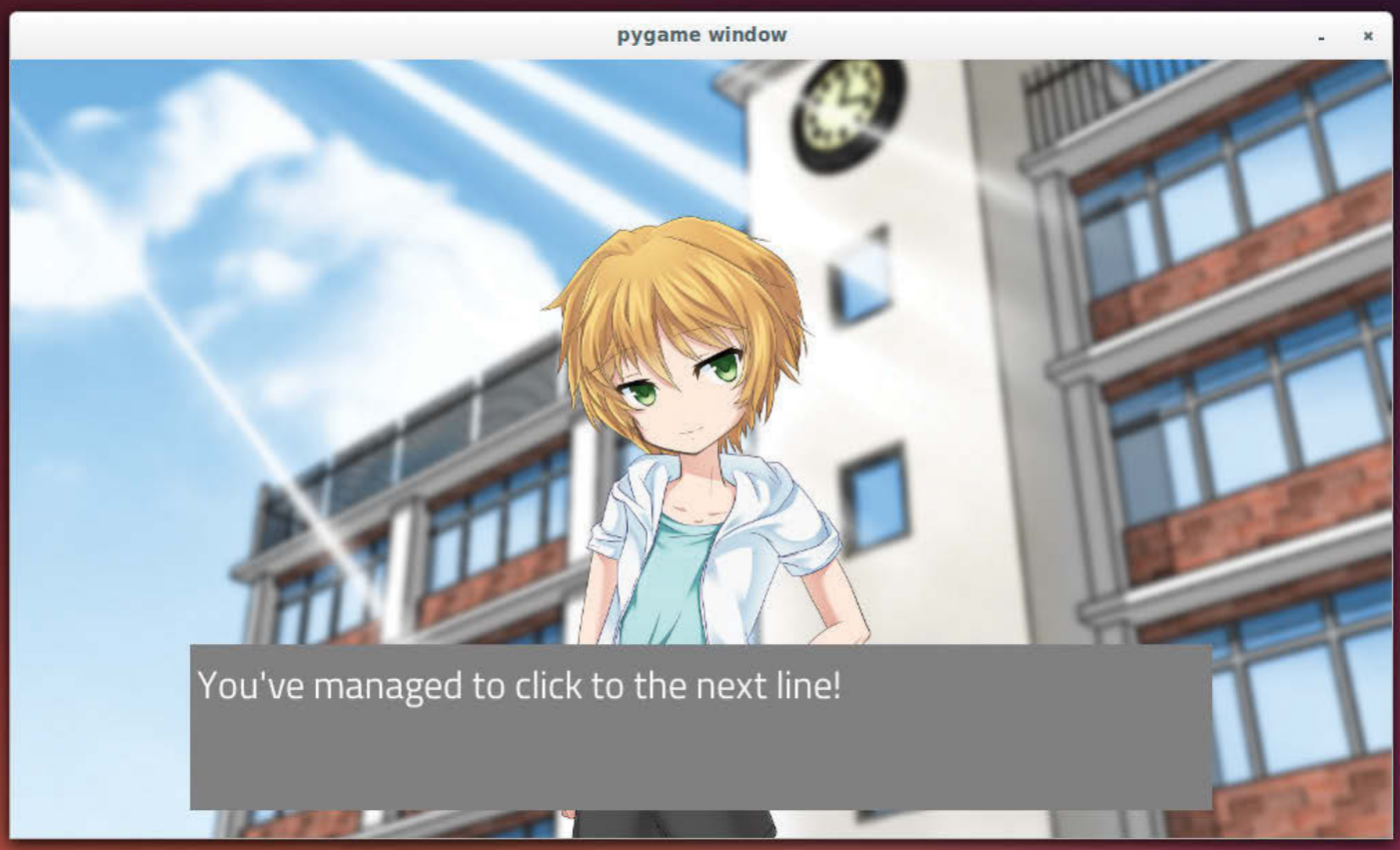
08 Starting the main game

We don't need many modules for the current state of our visual novel. Here we've imported the new pygame module, our script as a module and the time module for aesthetic reasons – we're going to have the code pause in bits rather than just instantly change scenes to the next line. We also initialise pygame with a simple pygame.init().

09 Add variables and assets

We can now add the mixture of info that we need to run the novel. We define the size of the display screen to use (1000 pixels wide and 563 high), along with some RGB colours for the code to use. We're also telling

Below Variables and assets are used to add interest visually and in the story



pygame which font to use, and how large for certain sections, and also loading images for the game.

10 Start the game

Create a display for the game. Pygame works by constantly updating the display with new information. To show how this works, the menu function adds elements to the display (which we've titled screen), such as filling it with colour, adding shapes and using blit to add images, or in this case text. Once you've created a buffer of changes to the screen, you update it with the flip() function.

11 See the mouse

As we've created the button as a rectangle and now an image on the menu, we need to know that the mouse is hovering over it to know when it's clicked the button. First we have to use event.get() to see the mouse in general, then we look for the position with get_pos(). After that, we wait for it to click, see where it clicked (using the co-ordinates of the rectangle) and then make a decision after that.

12 Start the story

Our start_game function is called when the mouse clicks the right position and we prepare the game, getting the characters, locations and progression through the game script. The rest of this function uses this info to pull in data from the script to make the game flow properly.

13 First screen

The first screen is handled differently to the rest, and acts to get every element up on the interface before we

"Pygame works by constantly updating the display with new information"

start continuing – it makes the code take up a little less time to process as we begin. The getattr allows us to use the string/integer associated with our place in the story and call upon the relevant scene function from the script file. We then use an if statement with an iterative function to successively add screen elements, to give the illusion that it's building up the first screen. We finish it by advancing the progression value.

Above Use an if statement with an iterative function to add screen elements, to give the illusion that it's building up the first screen

14 Go to the next line

Similar to how our original startup code works, our next if statement and iteration checks to see what is different on the next line, and if it moves to a different scene function, and changes anything that is different without filling up the buffer any more than is needed. We've made it so no change is labelled with a 0 in the scripts.

15 The starting function

We finish our code bit with a simple function that starts

off the entire game. This is just to encapsulate the entire code and allows us to add different ways of turning it off in the future. When running the file, IDLE will load everything up and then run the game() function at the end – this is similar to how you can add a __main__ function at the end that will start the code in the command line.

16 Expand your code

The code we've written is very expandable, enabling you to add decisions that are logged to take you to different scenes – or 'routes', in visual novel terminology – and make your game feel more interactive. This doesn't require much more code to the if statements, and is also a good way to look into adding graphical buttons to click in order to use the collide function.

17 Move the assets

Currently, the code has the script-specific assets in the main visualnovel file. These can be moved to the script, allowing you to make the visualnovel file much more modular so that can you have multiple scripts with different assets to load at startup.

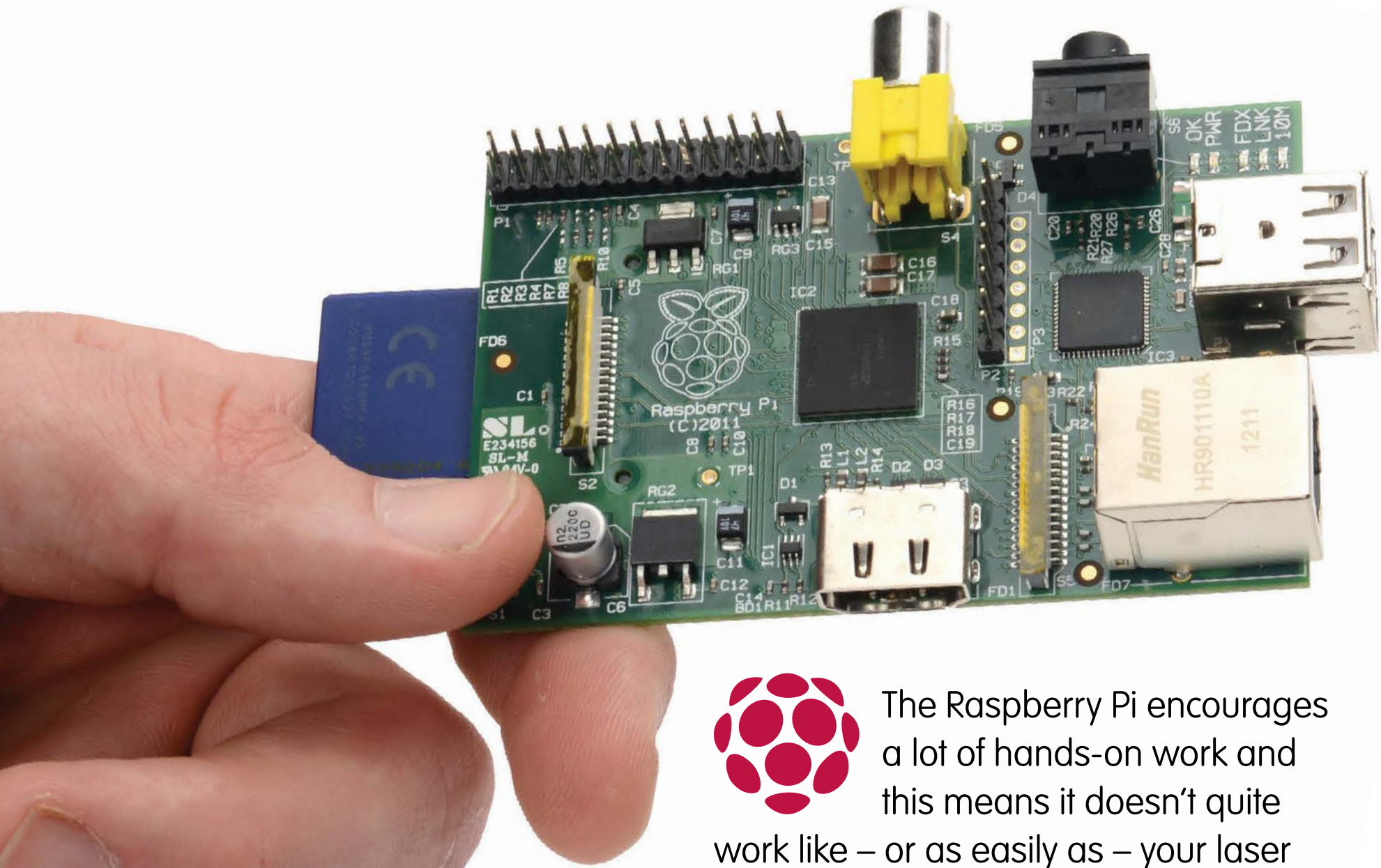
```
elif turn > 0 and click_state[0] == 1:
    line_start = 0
    for i in range (4):
        if line_start == 0 and line[0] != '0':
            print line[0]
            screen.blit(location[line[0]], [0, 0])
            time.sleep(1)
        elif line_start == 1 and line[1] != '0':
            screen.blit(character[line[1]], [377, 113])
            time.sleep(1)
        elif line_start == 2:
            pygame.draw.rect(screen, (grey), pygame.Rect(130, 423, 740, 120))
        elif line_start == 3:
            screen.blit(game_font.render(line[2], 1, white), (135, 430))
            turn += 1
            if  line[3] != '0':
                game_script = line[3]
                time.sleep(0.5)
            game_state = line[4]
            clicked = 0

        line_start += 1
        pygame.display.flip()
```

Left You can make the visual novel script more modular

Talking Pi

Join the conversation at...

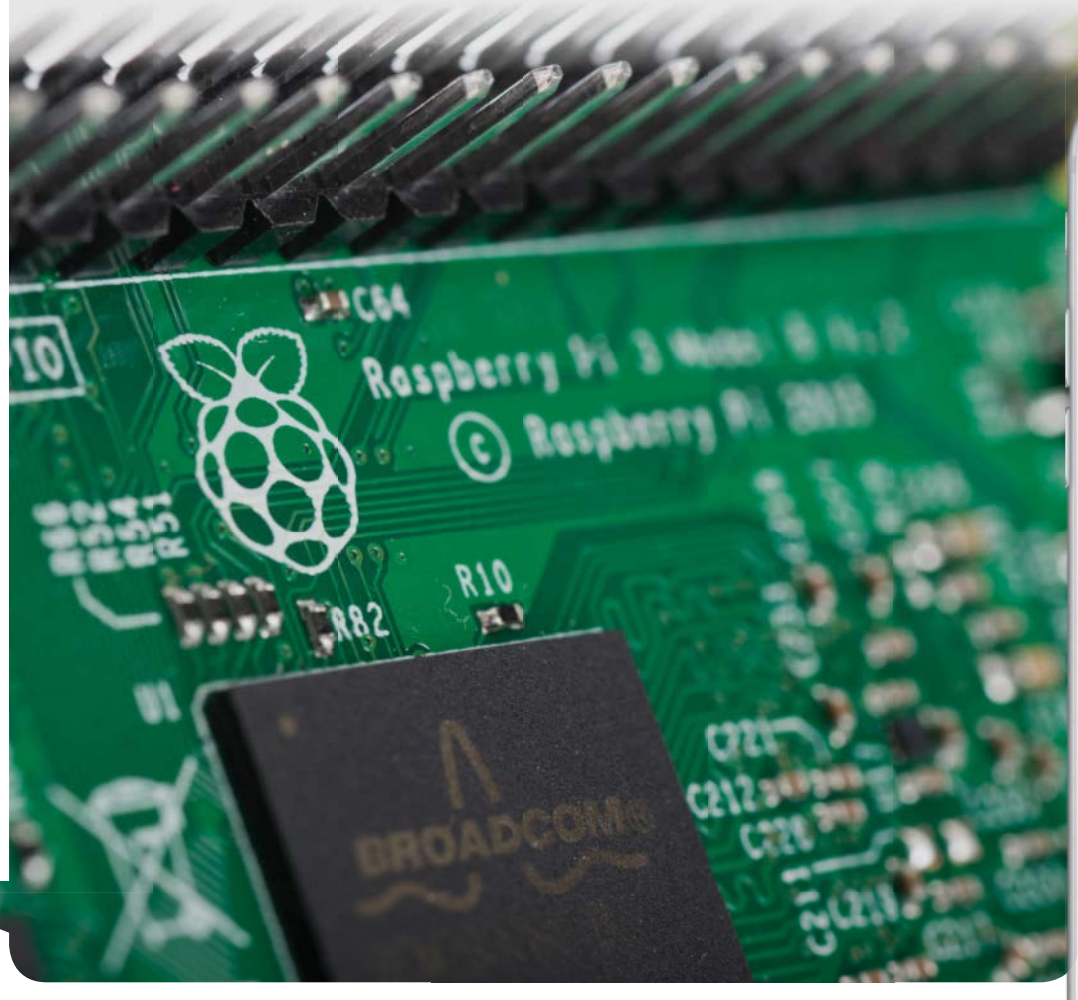
@linuxusermag | Linux User & Developer | RasPi@imagine-publishing.co.uk



The Raspberry Pi encourages a lot of hands-on work and this means it doesn't quite work like – or as easily as – your laser focus-tested smartphone interface, or even a normal computer. So we're answering your burning Raspberry Pi questions each issue – get in touch with us on Twitter, Facebook or by email.

Can I run Kodi on my Raspberry Pi, and will it be different from the version running on my other devices?
Gordon via email

Kodi is a great media player and it's absolutely possible to run it on the Raspberry Pi. At the time of going to press, the current stable version of Kodi for the Pi is Jarvis, which is exactly the same as the current stable version for Windows and Android (although it's worth noting that you need the version of Kodi designed for the Raspberry Pi – the Windows or Android version will not run on your Pi). You shouldn't notice any real difference, apart from the fact that it's better not to run Blu-Ray-quality video as it can slow down performance. Some people recommend disabling omxplayer acceleration to improve performance too.



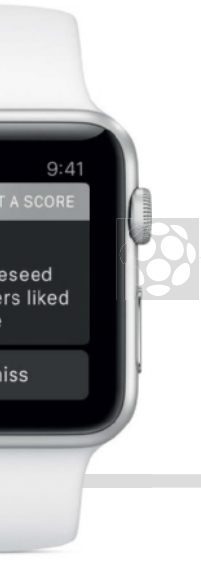
Keep up with the latest Raspberry Pi news by following @LinuxUserMag on Twitter. Search for the hashtag #RasPiMag

Which model of Pi is best for using in a classroom?
Mark via email

All Pis are excellent learning tools, but the one that the Raspberry Pi Foundation recommends using in schools is the Pi 2, because of how flexible it is – it works with first-generation Pi boards but is considerably more powerful, with a 900MHz quad-core ARM Cortex-A7 CPU and 1GB RAM. While it's not as cheap as the Pi Zero, being an older model it can be picked up for a bargain price. And don't forget that the Pi 2 is compatible with the micro:bit too!

What's the camera's maximum resolution?
Kelly via email

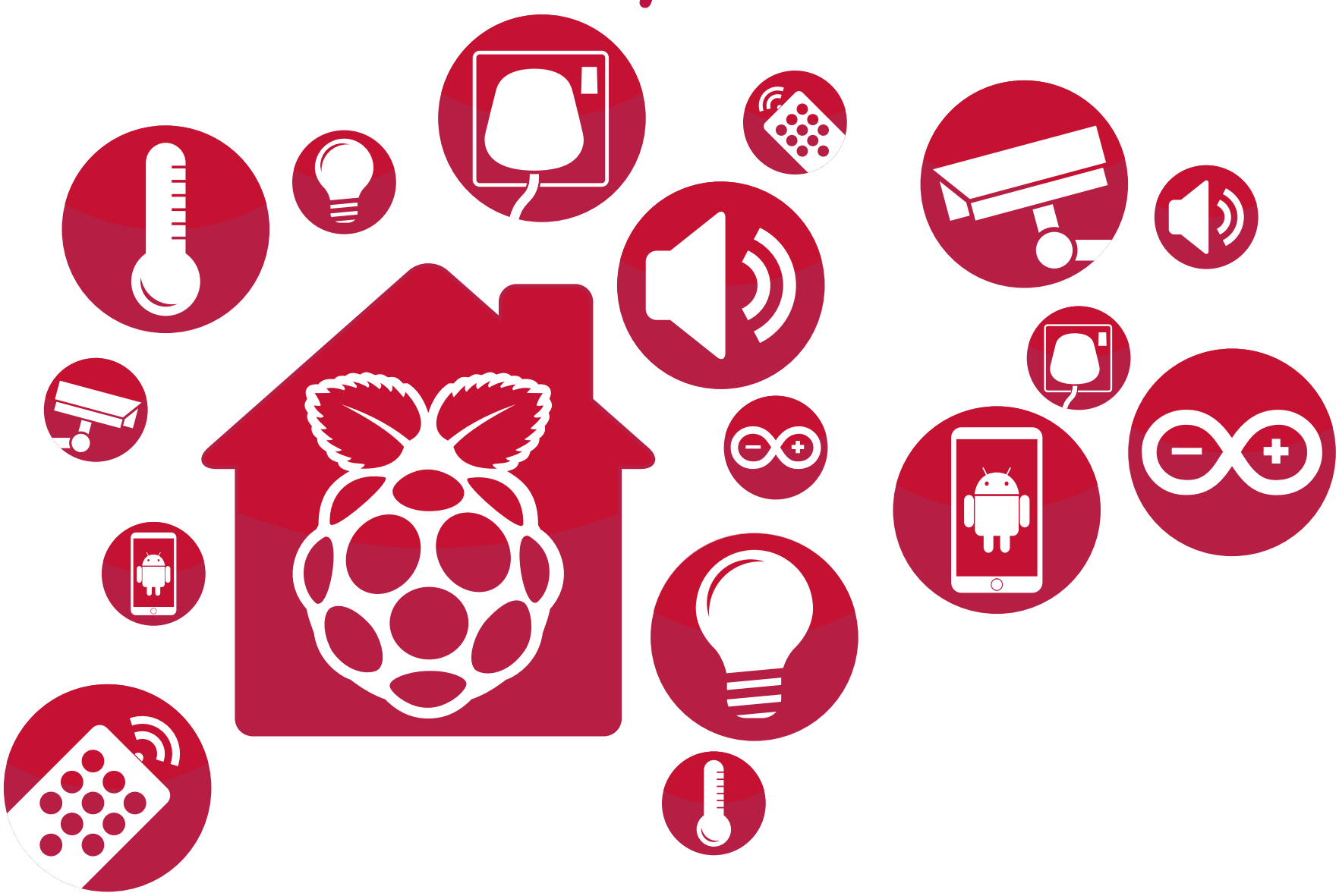
The Pi's Omnivision 5647 camera module is surprisingly powerful and easily comparable to the front selfie camera you'll find in the average mobile phone. It takes 5-megapixel photos at a size of 2592×1944 pixels, which is pretty respectable – certainly enough to print out a sharp image at normal photo print sizes, and even an A4 print will come out looking fairly crisp. It can also shoot RAW, the industry-standard lossless format – photos shot in this format will not lose data no matter how many times they're opened, closed, printed or transferred – as well as JPEG, PNG, GIF and BMP.



Next issue

Get inspired **Expert advice** **Easy-to-follow guides**

Home automation with your Pi



Get this issue's source code at:
www.linuxuser.co.uk/raspicode